

TECHNICKÁ UNIVERZITA v KOŠICIACH
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Ukázková aplikácia v softvéri
pre laboratórium virtuálnej reality
Diplomová práca

2017

Bc. Marián Hudák

TECHNICKÁ UNIVERZITA v KOŠICIACH
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Ukázková aplikácia v softvéri
pre laboratórium virtuálnej reality
Diplomová práca

Študijný program: Informatika
Študijný odbor: 9.2.1 Informatika
Školiace pracovisko: Katedra počítačov a informatiky (KPI)
Školiteľ: Ing. Štefan Korečko, PhD.
Konzultant: Ing. Štefan Korečko, PhD.

2017 Košice

Bc. Marián Hudák

Abstrakt v SJ

Diplomová práca sa zaoberá tvorbou aplikácie v systéme virtuálnej jaskyne nachádzajúceho sa na Technickej Univerzite v Košiciach, Fakulte počítačov a informatiky. Analýzou opisuje prvky systému, rozoberá nasadenie virtuálnych jaskýň, sprostredkúva pohľad na použitý softvér SSC SuperEngine. Prípravou trojrozmernej scény porovnáva výhody a nevýhody grafického vizualizačného jadra, aplikuje metodiky pre dosiahnutie reálneho vnemu v používaní vizuálnych efektov. Používaním rôznych štruktúr textúr a modelov v scéne hodnotí rýchlosť reakcií systému a porovnáva nároky kladené na výkon. Záverom hodnotí systém z pohľadu použiteľnosti a stability.

Kľúčové slová v SJ

Systém virtuálnej jaskyne, virtuálna realita, zmiešaná realita, 3D prostredie

Abstrakt v AJ

This thesis is concerned with designing an application in the system of virtual cave which is located at the Technical Univeristy of Košice, Department of Informatics and Computers. The analysis describes elements of the system, examines deployment of virtual caves in the world and interpretes view on used visualisation software from SSC SuperEngine. By preparation of three-dimensional scenes it compares pros and cons of visualisation core, applies methodics for reaching an impression of real sense by using visual effects. By using different structures of textures and models in scenes it rates a speed of reactions of the system and compares requirements demanding on performance. In conclusion the thesis rates system from the point of usability and stability.

Kľúčové slová v AJ

Virtual cave system, virtual reality, mixed reality, 3D enviroment

ZADANIE DIPLOMOVEJ PRÁCE

Študijný odbor: **Informatika**

Študijný program: **Informatika**

Názov práce:

Ukázková aplikácia v softvéri pre laboratórium virtuálnej reality

Demo application for Virtual Reality Laboratory Software

Študent: **Bc. Marián Hudák**

Školiteľ: **Ing. Štefan Korečko, PhD.**

Školiace pracovisko: **Katedra počítačov a informatiky**

Konzultant práce:

Pracovisko konzultanta:


Pokyny na vypracovanie diplomovej práce:

1. Oboznámiť sa s vizualizačným nástrojom SSC SuperEngine a VR jaskyňou dostupnou v laboratóriu LIRKIS FEI TUKE.
2. Analyzovať možnosti SSC SuperEngine a VR jaskyne a porovnať ich s podobnými existujúcimi riešeniami.
3. Navrhnuť a implementovať ukázkovú aplikáciu, ktorá by plne demonštrovala možnosti jaskyne a jej softvérového vybavenia.
4. Navrhnuť a implementovať rozšírenia vizualizačného nástroja, potrebné na realizáciu bodu 3.
5. Na základe experimentov s ukázkovou aplikáciou vyhodnotiť použiteľnosť VR jaskyne a jej vizualizačného nástroja.
6. Vypracovať dokumentáciu podľa pokynov vedúceho práce


Jazyk, v ktorom sa práca vypracuje: slovenský

Termín pre odovzdanie práce: 28.04.2017

Dátum zadania diplomovej práce: 31.10.2016


.....
doc. Ing. Jaroslav Porubán, PhD.
vedúci garantujúceho pracoviska



1.8. 
.....
prof. Ing. Liberios Vokorokos, PhD.
dekan fakulty

Čestné vyhlásenie

Vyhlasujem, že som celú diplomovú prácu vypracoval samostatne s použitím uvedenej odbornej literatúry.

Košice, 28.apríla 2017

.....

vlastnoručný podpis

Pod'akovanie

Veľké poďakovanie za spoluprácu a ochotu venujem Ing. Štefanovi Korečkovi, PhD. za profesionálny prístup vedený formou konzultácií rád a usmernení pri vytváraní diplomovej práce. Rovnako ďakujem Ing. Ladislavovi Jachovi za odbornú asistenciu a umožnenie prístupu do prostredia laboratória LIRKIS. Ďalej ďakujem rodine a priateľom za poskytnutie morálnej podpory pri zvládaní problematiky diplomovej práce.

Obsah

Zoznam obrázkov	10
Zoznam tabuliek	11
Slovník termínov.....	12
Úvod	13
1. Formulácia úlohy.....	15
2. Systém virtuálnej jaskyne TUKE FEI.....	16
2.1. Virtuálna jaskyňa	16
2.1.1. Prvky systému	16
2.1.2. Využitie CAVE systému	19
2.1.3. Interakcie v systéme.....	19
2.2. Systém SuperEngine	20
2.2.1. Komponenty systému	20
2.2.2. Vizualizačné jadro OpenSG.....	22
2.2.3. Importovanie a exportovanie scén	23
2.2.4. Renderovanie scén	23
2.2.5. Statické objekty scény	25
2.2.6. Dynamické objekty scény	25
2.2.7. Svetelné prvky	26
2.2.8. Zvukové prvky.....	26
2.3. Praktické dôsledky využívania systému	27
2.3.1. Kompozícia scény	27
2.3.2. Simulácia scény	28
2.3.3. Limity systému.....	29
2.3.4. Interakcie s používateľom	30
2.4. Nasadenie virtuálnych jaskýň.....	30
2.4.1. Simulačné systémy	31
2.4.2. Edukačné prostredia	32

2.4.3.	Výskum	33
2.4.4.	Zábava a voľný čas.....	34
2.5.	Trojrozmerná scéna virtuálnej jaskyne	35
2.5.1.	Charakteristika scénických prvkov	35
2.5.2.	Voľba vlastností kamery	37
2.5.3.	Transformácie objektov a uzlov	38
2.5.4.	Výber svetelných prvkov	39
2.5.5.	Okolie prostredia.....	40
3.	Návrh aplikácie v prostredí SuperEngine	41
3.1.	Hlavné aspekty návrhu prostredia	41
3.1.1.	Postup overenia funkcionality systému virtuálnej jaskyne	42
3.1.2.	Úlohy aplikácie v prostredí virtuálnej jaskyne.....	42
3.1.3.	Rozdelenie scény	44
3.1.4.	Graf scény.....	45
3.1.5.	Materiály objektov	45
3.1.6.	Svetelné prvky scény	49
3.1.7.	Interakcie objektov scény.....	51
3.2.	Realizácia skriptovania scény	52
3.2.1.	Základná konštrukcia skriptu.....	52
3.2.2.	Sekcie skriptu riadenia scény	53
3.2.3.	Modulárne viazanie skriptov	54
3.2.4.	Vplyv nesprávnej aplikácie skriptu	55
4.	Implementácia aplikácie scény v prostredí virtuálnej jaskyne.....	56
4.1.	Zobrazenie prostredia virtuálneho sveta	57
4.1.1.	Scéna virtuálneho robotického ramena.....	57
4.1.2.	Scéna kokpitu lietadla	59
4.1.3.	Scéna halového žeriavu.....	62
4.2.	Využitie vstupov systému.....	64

4.2.1.	Overenie vstupov pre klávesnicu	65
4.2.2.	Overenie vstupov pre myš	66
4.2.3.	Overenie vstupov pre Joystick.....	66
4.2.4.	Overenie vstupov pre Gamepad	68
4.2.5.	Overenie vstupov pre OptiTrack	69
4.2.6.	Aplikovanie interaktívnych tlačidiel	71
4.3.	Dosiahnutie reálnejšieho vnemu	72
4.3.1.	Aplikácia pečenia textúr	72
4.3.2.	Dosadenie Skybox objektov	73
4.3.3.	Rozšírenie spätnej väzby	74
5.	Vyhodnotenie aplikácie systému SuperEngine	75
5.1.	Porovnanie výsledných funkcionalít.....	75
5.1.1.	Porovnanie vplyvu zložitosti 3D modelov	76
5.1.2.	Porovnanie vplyvu textúr	77
5.2.	Faktory ovplyvňujúce reakcie systému	79
5.2.1.	Vplyv postupu pri spúšťaní systému	80
5.2.2.	Vplyv konfigurácie vstupných zariadení.....	80
5.2.3.	Vplyv nesprávnej konštrukcie skriptu	81
6.	Záver.....	82
	Zoznam použitej literatúry	83
	Prílohy	86

Zoznam obrázkov

Obr. 1. Porovnanie typizovaného a netypizovaného tvaru.....	17
Obr. 2. Virtuálna jaskyňa Technickej Univerzity vo Zvolene.	18
Obr. 3 Zloženie systému SuperEngine.....	22
Obr. 4 Štruktúra balíčka scény.	23
Obr. 5 Rozdelenie úloh vykreslenia obrazu.....	24
Obr. 6 Názorná ukážka grafu scény.....	28
Obr. 7 Simulácia stavebného vozidla v prostredí virtuálnej jaskyne. [22]	32
Obr. 8 Edukačný systém pre kompozíciu palivových článkov NASA. [24].....	33
Obr. 9 Aplikácia prostredia virtuálnej jaskyne v medicínskom výskume. [27].....	34
Obr. 10 Posunutie kamery v smere osi y s bodom (A) a bez bodu orientácie pohľadu (B).	38
Obr. 11 Spôsoby viazania objektov pre aplikáciu transformácií.	39
Obr. 12 Graf scény s obsahom modulov.	45
Obr. 13 Prispôsobenie siete modelu v modelovacom nástroji.	46
Obr. 14 Prispôsobenie textúry v grafickom nástroji.....	47
Obr. 15 Aplikácia siete s detailným rozdelením plôch objektu.....	47
Obr. 16 Aplikácia siete s jednoduchým rozdelením plôch objektu.....	48
Obr. 17 Neprirodzené šírenie svetla v priestore.	50
Obr. 18 Porovnanie scény s generovaním svetla vizualizačného jadra a svetelnou textúrou.	51
Obr. 19 Scéna s robotickým ramenom.....	58
Obr. 20 Graf scény kinematického ramena.....	59
Obr. 21 Graf scény kokpitu lietadla.....	60
Obr. 22 Model kokpitu lietadla Boeing 737 v systéme virtuálnej jaskyne.	61
Obr. 23 Halový žeriav v SuperEngine.	62
Obr. 24 Graf scény žeriavu.	63
Obr. 25 Zobrazenie komunikácie externej aplikácie určenej pre zariadenie Joystick.....	67
Obr. 26 Používanie zariadenia Joystick v systéme virtuálnej jaskyne.	68
Obr. 27 Zobrazenie komunikácie externej aplikácie určenej pre zariadenie Gamepad.	68
Obr. 28 Používanie zariadenia Gamepad v systéme virtuálnej jaskyne.....	69
Obr. 29 Rozdiely dosahované pri používaní systému OptiTrack.....	70
Obr. 30 Interaktívne tlačidlá scény halového žeriavu.	71
Obr. 31 Scéna v prostredí SuperEngine s pečenými textúrami.....	73
Obr. 32 Vplyv zložitosti štruktúr 3D modelov na plynulosť vizualizácie scény.....	77
Obr. 33 Vplyv zložitosti štruktúr textúr na plynulosť vizualizácie scény.	79

Zoznam tabuliek

Tab. 1 Porovnanie FPS vplyvu formátov 3D modelov.	76
Tab. 2 Porovnanie vplyvu textúr s formátom modelu *.3ds	78
Tab. 3 Porovnanie vplyvu textúr s formátom modelu *.obj	78
Tab. 4 Porovnanie vplyvu textúr s formátom modelu *.fbx	78
Tab. 5 Porovnanie vplyvu textúr s formátom modelu *.wrlm.	79
Tab. 6 Faktory problémov konfigurácie zariadení.....	80
Tab. 7 Faktory chýb v konštrukcii skriptu.....	81

Slovník termínov

LIRKIS CAVE	Laboratórium inteligentných rozhraní komunikačných a informačných systémov na Technickej univerzite v Košiciach, obsahujúce systém virtuálnej jaskyne pre výskum v oblasti virtuálnej reality.
Sledovanie pohybu	(z ang. Tracking) je systém charakteristický pre sledovanie bodov v ohraničenom priestore, používaný najčastejšie v systémoch virtuálnych jaskýň alebo prenášaním pohybu reálne snímaných bodov viazaním na trojrozmernú štruktúru.
Slovakia SuperComputers	(v skratke SSC) je spoločnosť podieľajúca sa na tvorbe systému virtuálnej jaskyne v laboratóriu virtuálnej reality na Technickej univerzite v Košiciach. Zameranie spoločnosti je orientované vo vývoji systémov virtuálnej reality, implementácii 3D softvéru, tvorby 3D vizualizácií, aplikácií 3D skenovania.
Pečenie textúr	(z ang. Baking Textures) reprezentuje jednu z inovatívnych metód pri riešení problematiky nárokov svetelnej scény v prostredí virtuálnej reality. Metóda je často aplikovaná generovaním textúry s obsahom svetlenej vrstvy materiálu aplikovaného na 3D objekte formou textúry.
Snímková frekvencia	(z ang. Frame Per Second / FPS) reprezentuje počet celých obrázkov zobrazených za jednu sekundu na zobrazovacom grafickom zariadení. Vnímanie ľudského oka dokáže plynulo sledovať frekvenciu 24 obrázkov za jednu sekundu.

Úvod

Náplňou diplomovej práce je vytvorenie ukázkovej aplikácie v systéme virtuálnej jaskyne SuperEngine, umiestnenej na Technickej univerzite v Košiciach, Katedre počítačov a informatiky. Hlavná úloha predstavuje prípravu trojrozmernej scény s využívaním dostupných prostriedkov systému a metód pre vytváranie rôznych druhov interakcií, rovnako transformácií v trojrozmernom prostredí scény, overení podpory z pohľadu trojrozmerných štruktúr a dosahovaní reálnejšieho vnemu z pohľadu používateľa.

Očakávaným výsledkom práce je hodnotenie systému prostredníctvom praktických skúseností nadobudnutých pri testovaní vytvorených scén zameraných na rôzne funkcionality systému z pohľadu vývojára a rovnako používateľa. Existencia systémov virtuálnych jaskýň zasahuje do začiatku 90.rokov, v ktorých využívaním jednoduchej konfigurácie zobrazovacích zariadení bol obraz prenášaný na steny miestnosti tvaru kocky alebo kvádra. Tvar a veľkosť virtuálnych jaskýň sa v priebehu času menili a zlepšovali na vyššiu úroveň. Vplyvom vývoja hardvéru sa rovnako zdokonaľovala softvérová stránka z pohľadu docielenia plynulejších a zložitejších efektov, podpory ovládacích zariadení, zobrazovanie scény dosahovalo zložitejšiu úroveň s vizuálnym efektom blízky reálnemu prostrediu. Vrchol zdokonaľovania virtuálnych jaskýň nie je známy, budúcnosť týchto systémov je stále otvorená a predstavuje širokú škálu používania v rôznych odvetviach výskumu, edukácie alebo simulačných technológií. Dnešné systémy virtuálnych jaskýň využívajú rôzne vizuálno-zvukové efekty, časticové systémy, interakcie používateľa a systému, dynamické prvky scény. Každý z efektov závisí od možností vizualizácie scény podľa dostupných prostriedkov vizualizačného jadra, ktorého úlohou je vizualizácia trojrozmerného prostredia aplikovaním štruktúr trojrozmerných objektov a materiálov. Najčastejším problémom spomínaných systémov je zložitá konfigurácia v dôsledku používania rôznych spôsobov zobrazovania prostredia. Najznámejšie aplikácie systémov virtuálnych jaskýň využívajú fyzické prostredie tvaru kocky s premietaným obrazom na stenách a rozdelením úloh vykresľovania vo výpočtovom klastri, obsahujúcom jednotlivé výpočtové jednotky zobrazujúce scénu podľa pridelenej časti v systéme virtuálnej jaskyne. Systém LIRKIS CAVE je konštrukčne komplexnejší oproti jednoduchým systémom reprezentovaných prostredím kocky. Systém využíva 20 obrazoviek s panoramatickým zobrazením scény v tvare polkruhu, v rozmedzí 180 stupňov, ktorý dovoľuje používateľovi vstúpiť do prostredia s dosahovaním výraznejšieho dojmu dostupnosti trojrozmerných objektov v ňom obsiahnutých. Z pohľadu softvérového odbavenia vizualizovania scén je v systéme aplikovaný SuperEngine od spoločnosti Slovakia SuperComputers. Softvér SuperEngine umožňuje vytváranie scén pomocou viazania skriptov a trojrozmerných štruktúr s využívaním dostupných metód pre aplikovanie rôznych transformácií v priestore scény,

interakcií medzi objektmi a práce s kamerou vo virtuálnom prostredí. Úlohou diplomovej práce je docielenie vytvorenia scény s aplikovaním rôznych kombinácií spomínaných metód a rozšírenia podpory vstupných zariadení v systéme. Vizualizačné jadro systému na báze OpenGL z roku 2006 je v porovnaní s aktuálnymi typmi vizualizačných jadier výrazne obmedzené. Rovnako dôležité je overenie rôznych metodík pre dosahovanie reálneho vnemu z pohľadu používateľa a porovnanie nárokov kladených scénou na výkon výpočtového systému. Pomocou porovnávania zložitostí štruktúr 3D objektov a textúr je možné dosiahnuť optimálnu úroveň efektov trojrozmernej scény, pri ktorých scéna dosahuje výrazne lepší vizuálny dojem a funkcionálnosť.

Osobne považujem riešenie problému testovania scény s dosahovaním reálnejšieho vnemu a vyššej úrovne efektov za reálne a dosiahnuteľné. Pre používanie rôznych metodík tvorby graficky náročnej scény je dôležité vyčlenenie limitov systému. Prvý bod realizácie predstavuje získavanie informácií o systéme, konfigurácií zariadení a komponentov systému, zameranie sa na zistenie spôsobu komunikácie a riadenia úloh systému v počítačovom klastri. Druhý bod sa zaoberá získavaním informácií o skriptovaní s používaním skriptovacieho jazyka Ruby a knižníc vytvorených spoločnosťou Slovakia SuperComputers. Rovnako je v druhom bode obsiahnuté samotné modelovanie a skriptovanie pre testovanie plynulosti zobrazenia scény a orientácie kamery. Tretím bodom je realizované dosahovanie vyššej úrovne efektov prostredníctvom vytvárania štruktúrne zložitejšej scény a používaním metodík v dosahovaní reálnejšieho vnemu z pohľadu používateľa. Podľa možností komunikácie so systémom SuperEngine je rovnako realizované testovanie rozšírenia vstupných zariadení pre doposiaľ neobsiahnuté vstupy s overením ich presnosti a využitia v trojrozmerných scénach. Predpokladom aplikácie uvedených bodov je vyhodnotenie schopností systému virtuálnej jaskyne s uplatnením využívania trojrozmerných prostredí na úrovňovo vyššej vizuálnej stránke a simulácie. Aplikáciou metodík pre dosahovanie prirodzenejšie pôsobiacej scény, s využitím externého modelovacieho nástroja, považujem za jedno z možných riešení problému úlohy pre uľahčenie výpočtových nárokov kladených na systém. Prihliadnutím na dlhodobé osobné skúsenosti pri práci s trojrozmernými štruktúrami, scénickým modelovaním, skriptovaním trojrozmerných prostredí, považujem problém za riešiteľný.

1. Formulácia úlohy

Problematika diplomového projektu je zameraná na návrh a tvorbu aplikácie v systéme virtuálnej jaskyne dostupnej v laboratóriu LIRKIS FEI TUKE v Košiciach, pre výskum v oblasti virtuálnej reality. Zámerom analýzy je rozdelenie jednotlivých častí systému SSC SuperEngine a opisu ich podstaty pri riadení simulácie scény. Porovnávaním fyzickej konštrukcie LIRKIS CAVE opisuje rôzne typy virtuálnych jaskýň vo svete vrátane ich zamerania v rôznych aplikáciách. Ďalším delením systému porovnáva aktuálnu verziu vizualizačného jadra s podobnými existujúcimi riešeniami používanými vo svete. Využívaním trojrozmerného virtuálneho prostredia a dostupných prostriedkov systému SSC SuperEngine aplikuje vytváranie scén prostredníctvom implementácie skriptov vrátane dosahovania interakcií s používateľom. Využívaním rôznych typov štruktúr trojrozmerných objektov aplikuje jednoduché a zložené transformácie pre klasifikovanie použiteľnosti dynamických vlastností scén. Aplikovaním trojrozmerných modelov v scénach s rozdielnou zložitou štruktúrou testuje použiteľnosť systému z pohľadu nárokov kladených na výkon LIRKIS CAVE. Implementáciou vstupných zariadení rozširuje podporu hardvérových prvkov pre ovládanie objektov virtuálneho prostredia. Rozšírením spätnej väzby viazanej na hardvér zdokonaľuje komunikáciu medzi používateľom a systémom. Výsledným hodnotením využíva experimentálne overenie jednotlivých funkcionalít systému, s porovnávaním rýchlostí reakcií pri rôznych úrovniach záťaže a plynulosti simulácie scény. Záverom je systém klasifikovaný podľa dosahovania reálnejšieho vnemu scény, reakcií na požiadavky používateľa, efektov dosiahnuteľných v scéne, podpory rozšírenia ovládacích zariadení.

2. Systém virtuálnej jaskyne TUKE FEI

Systémy virtuálnych jaskýň poskytujú druh zmiešanej reality pre používateľov z rôznych aspektov využitia v rámci aplikácie simulácií, edukácie alebo zábavy. V uvedenej kapitole sa pozrieme na LIRKIS CAVE z pohľadu vnútornej štruktúry a komunikácie medzi jednotlivými prvkami riadenia scény virtuálneho prostredia. Z pohľadu využitia kapitola uvádza aplikovanie a viazanie hardvérových a softvérových prvkov, prepojenie medzi reálnym a virtuálnym priestorom. Z pohľadu analýzy existujúceho systému rozoberá druh interakcie medzi používateľom a systémom pomocou využívania prvkov sledovania pohybu v priestore virtuálnej jaskyne. Prostredníctvom existujúcich systémov používaných v ostatných krajinách porovnáva jednotlivé uplatnenia v sfére simulačných systémov na úrovni používania špecifických hardvérových prvkov, systémov pre edukáciu a odborné využitie v rôznych odvetviach. Pomocou rozdelenia systému rozoberá jednotlivé komponenty riadenia systému a komunikácie medzi nimi. Kapitola podobne rozoberá jednotlivé uplatnenia systémov virtuálnych jaskýň v rôznych odvetviach. Pomocou charakteristiky scénických prvkov kategorizuje použitie rôznych typov objektov scény vrátane kamery v aktuálnom systéme. Záverom kapitola dekomponuje porovnáva aplikácie transformácií vo vzťahu transformačných uzlov scény.

2.1. Virtuálna jaskyňa

Pojem Virtuálna Jaskyňa (CAVE) využíva typ zmiešanej reality, ktorý umožňuje fyzicky umiestniť osobu reálneho sveta do virtuálneho prostredia. Hlavnou úlohou virtuálnej jaskyne je vytváranie virtuálneho prostredia s možnosťou dosadenia fyzicky prítomného človeka v jej priestore. Druh interaktivity závisí od formy simulovaného prostredia a spôsobu reakcií systému na jednotlivé podnety zo strany používateľa. Každý CAVE systém obsahuje komponenty, ktoré realizujú sprístupnenie virtuálneho sveta prostredníctvom grafického výstupného zariadenia, rovnako dopĺňujú scénu zvukovým efektom v rôznych častiach prostredia. Systém je plne imerzívny, dosahujúci realistický zážitok pri vizualizovaní scény. Podobne sú v systéme obsiahnuté prvky senzorického snímania pohybu alebo zmeny niektorej zo sledovaných vlastností používateľa (pohyb, gestikulácia). Výsledok realizuje riadiaci systém, prevádzkujúci transformáciu akcie používateľa na reakciu systému, zabezpečujúci primeranú interakciu medzi používateľom a virtuálnym prostredím jaskyne.

2.1.1. Prvky systému

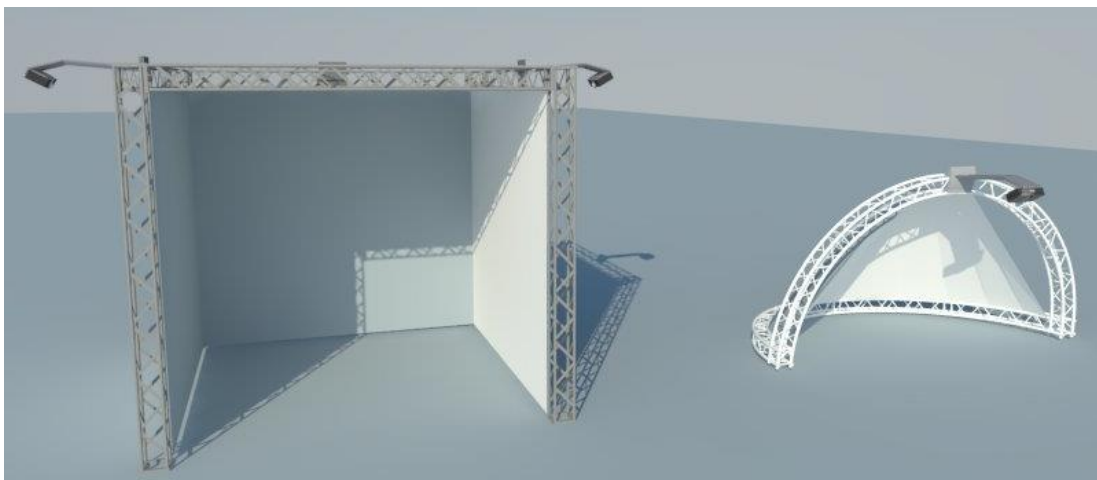
Systém virtuálnej jaskyne reprezentujú prvky podieľajúce sa na realizácii simulácie virtuálneho sveta s vykonávaním určitého druhu funkcionality. Podľa Burdea [1] každý CAVE

system využívá zobrazovacie metódy pre realizáciu virtuálneho prostredia v reálnom priestore s prispôsobeným tvarom a deformáciou obrazu podľa výsledného tvaru jaskyne [Obr.1].

Medzi hlavné komponenty reprezentujúce prvky virtuálnej jaskyne sú radené:

- konštrukčná časť,
- zobrazovacia časť,
- komunikačná časť,
- riadiaca časť.

Konštrukčná časť systému virtuálnej jaskyne reprezentuje základný element, prvotný bod realizácie systému, podľa ktorého sú dimenzované ostatné komponenty. Tvar virtuálnych jaskýň je reprezentovaný základnou konštrukciou v tvare kocky alebo kvádra so stenami používanými pre premietanie obrazu virtuálneho sveta, najčastejšie pomocou projektorov rozmiestnených v okolí stien konštrukcie. V opačnom prípade môže byť konštrukcia jaskyne prispôbena tvaru virtuálneho prostredia alebo potrebnej funkcionalite pre vytváranie reálnejšieho dojmu z pohľadu používateľa. Charakteristickým príkladom reprezentujúcim rozdiely medzi virtuálnymi jaskyňami je bežný tvar kvádra, polkruhu alebo polgule. Atypické tvary reprezentujú rôzne špecifické použitie, napríklad simulačné kabíny, operačné prostredia v rámci armádnych účelov alebo lekárske výskumy, prípadne deformácie primitívnych trojrozmerných útvarov.



Obr. 1. Porovnanie typizovaného a netypizovaného tvaru.

Základom pre vytváranie obrazového vnemu je zobrazovacia časť zložená z rôzneho počtu zobrazovacích prvkov, zariadení pre vysielanie a zachytávanie obrazu. Najčastejšie riešenie, poskytujúce reálnejší vnem, využíva zobrazovanie prostredníctvom projekčných zariadení s využitím 3D obrazu, stereoskopického zobrazenia [2]. 3D zobrazenie môže byť aplikované pomocou delenia obrazu s využívaním dvoch projektorov aplikovaných na jedno zobrazovacie

pole, alebo funkciou premietania s využitím moderných technológií, ktoré poskytujú automatické generovanie 3D obrazu, delením pre pravé a ľavé oko (LCD monitory a televízory, inteligentné 3D projektory) [Obr.2]. Komunikácia využívaná v systémoch virtuálnych jaskýň predstavuje spôsob interaktivity využívaný medzi viacerými používateľmi umiestnenými buď v jednej alebo viacerých virtuálnych jaskyniach v reálnom čase. Bežný spôsob komunikácie medzi používateľmi nasadenými v reálnom čase v rámci jednej virtuálnej jaskyne pozostáva z bežnej slovnej a fyzickej komunikácie. V prípade využívania viacerých virtuálnych jaskýň v reálnom čase je potrebné zabezpečiť postačujúci systém komunikujúci sieťovým pripojením pre prenos jednotlivých informácií medzi používateľmi v reálnom čase. Každý z používateľov vlastní vo virtuálnom prostredí avatar, pod ktorým vystupuje medzi ostatnými používateľmi. Rozdelenie riadenia systému je definované z dvoch pohľadov, a to z pohľadu správania používateľa ovládajúceho systém rôznymi prvkami alebo riadením komunikácie medzi jednotlivými komponentmi systému [3]. Každý systém využíva riadiace komponenty, ktoré zabezpečujú transformáciu komunikačného toku a docieľujú prostredníctvom akcie používateľa vytvorenie reakcie zo strany systému. V skutočnosti existuje niekoľko systémov sledovania pohybu alebo gestikulácie používateľa, ktoré umožňujú sledovať pohyb v reálnom čase a rozlíšiť definovanú, vloženú gestikuláciu. Najpopulárnejším trendom k docieleniu správnej interakcie používateľa a prostredia virtuálnej jaskyne je snímanie umiestnených bodov na tele používateľa prostredníctvom aplikácie infračervenej kamery a konfigurácie výrazných bodov v priestore. Takýto systém môžeme považovať za vstupný systém riadenia virtuálneho prostredia jaskyne.



Obr. 2. Virtuálna jaskyňa Technickej Univerzity vo Zvolene.

2.1.2. Využitie CAVE systému

Virtuálne prostredia predstavujú prínos v rôznych vedných odboroch a oblastiach využívajúcich simuláciu alebo vizualizáciu objektov, štruktúr, udalostí alebo prezentujú rôzne vlastnosti prostredníctvom interaktívneho virtuálneho sveta. Každé využitie umožňuje vytvárať priestor, ktorého hranice predstavujú výkonnostné hranice systému. Výhodou využívania CAVE systémov je predovšetkým možnosť implementácie rôznych prvkov v rámci ovládania (hardvér) a viazanie funkcionality prostredníctvom virtuálneho sveta (softvér). Kombináciou hardvérových prvkov a virtuálneho prostredia je možné prenášať rôznu funkcionality v reálnom čase a prispôbovať rôzne vlastnosti virtuálneho sveta pre hardvérové prvky. Pohľadom Shermana [4] Aplikácia CAVE systémov reprezentuje otvorenú budúcnosť s prínosom reálnejšieho vnemu prostredníctvom modernejších obrazových technológií. Dôležitosť využívania CAVE systému predstavuje vytváranie klamlivého vnemu na používateľa z pohľadu prelínania reálneho a virtuálneho sveta, v ktorom používateľ nepozná rozdiel medzi skutočným a virtuálnym priestorom. V takomto prípade je systém virtuálnej jaskyne užitočný, so splnením požiadaviek, pre ktoré bol určený. CAVE systém môže rovnako predstavovať nevýhodu v prípade nevhodnej kombinácií technológií alebo konfigurácie prostredia pre používateľa. Dezorientácia v priestore alebo nevoľnosť pri používaní systému je zaznamenaná v minimálnom počte, pre zdokonalenie systému je vhodné takéto prípady eliminovať a zabezpečiť vhodné reakcie prostredia na podnety používateľa správnym konfigurovaním interakcie systému.

2.1.3. Interakcie v systéme

Vzťah medzi používateľom a systémom je možné vyjadriť prostredníctvom interakcií vznikajúcich medzi oboma stranami v priebehu času vykonávania aktivít. Interakcia systému reprezentuje primeranú reakciu na podnet zo strany používateľa a zabezpečuje primeraný chod v priebehu používania systému. Systém virtuálnej jaskyne je interaktívny v prípade, ak dokáže spracovávať vstupy vytvorené používateľom bez oneskorenia reakcií alebo vykonania nesprávnych krokov. Virtuálna jaskyňa LIRKIS CAVE disponuje systémom interakcií, aplikáciou sledovania bodov (markers) pomocou infračervených kamier a prenášaním gestikulácií používateľa do prostredia virtuálneho sveta (tracking). V prípade nasadenia viacerých osôb do virtuálneho prostredia systém dokáže sledovať len jedného používateľa bez špecifického delenia bodov (markers). Takýto prípad rozdeľuje používateľov v reálnom čase na jedného interaktívneho a ostatných, sledujúcich interakcie medzi systémom a označeným používateľom. Systém obsahuje niekoľko druhov komunikácie zastupujúcu vizuálne interakcie v dôsledku zmeny alebo transformácie prostredia, zvukové interakcie viazané na objekty prostredia alebo časti scény [5]. Vizuálne interakcie zahŕňajú rôzne typy transformácií scény pri zmene polohy používateľa alebo jeho pohybu

v prostredí jaskyne, s využitím zmeny rýchlosti pohybu scény synchronne s pohybom používateľa. Zvukové interakcie rozširujú scénu o interaktívnejšie prostredie a rovnako vnem viazaným na časti aplikovanej scény alebo objekty v nej vystupujúce. Každá interakcia môže byť voliteľná, staticky určená alebo dynamická, celá množina interakcií závisí od druhu prostredia a úlohy, ktorá je od používateľa očakávaná.

2.2. Systém SuperEngine

Systém virtuálnej jaskyne aplikovaný na Fakulte Elektrotechniky a Informatiky Technickej Univerzity v Košiciach disponuje softvérom SuperEngine od spoločnosti Slovakia Super Computers [6], ktorý realizuje vizualizáciu trojrozmernej scény a prostredníctvom skriptovania v rámci jazyka Ruby [7] umožňuje komponovanie dynamickej scény. Využívanie skriptovacieho jazyka rozširuje použiteľnosť jaskyne pre rôzne trojrozmerné scény, rovnako podporuje implementáciu rôznych vstupných zariadení systému. Vykresľovanie alebo renderovanie scén poskytuje škálu pre prednastavenú konfiguráciu scény z pohľadu svetelných podmienok, povrchových materiálov, štruktúr obsahujúcich špecifickú vlastnosť vrátane odlesku alebo transparentnosti. Scéna zahŕňa statické a dynamické objekty s možnosťou vytvárania interakcií na podnety používateľa alebo dynamických prostredí s aplikáciou generovania javov. Detailné zobrazenie prispieva k reálnejšiemu vnemu vrátane tieňovania alebo pridávania efektov scény s aplikáciou svetelných prvkov, dotvárajúcich prirodzenejšie prostredie. Prioritným je rovnako používanie zvukových efektov viazaných na objekty alebo časti scény, ktoré je možné aplikovať nezávisle na usporiadaní, prípadne používať lokálne v rámci jednej scény.

2.2.1. Komponenty systému

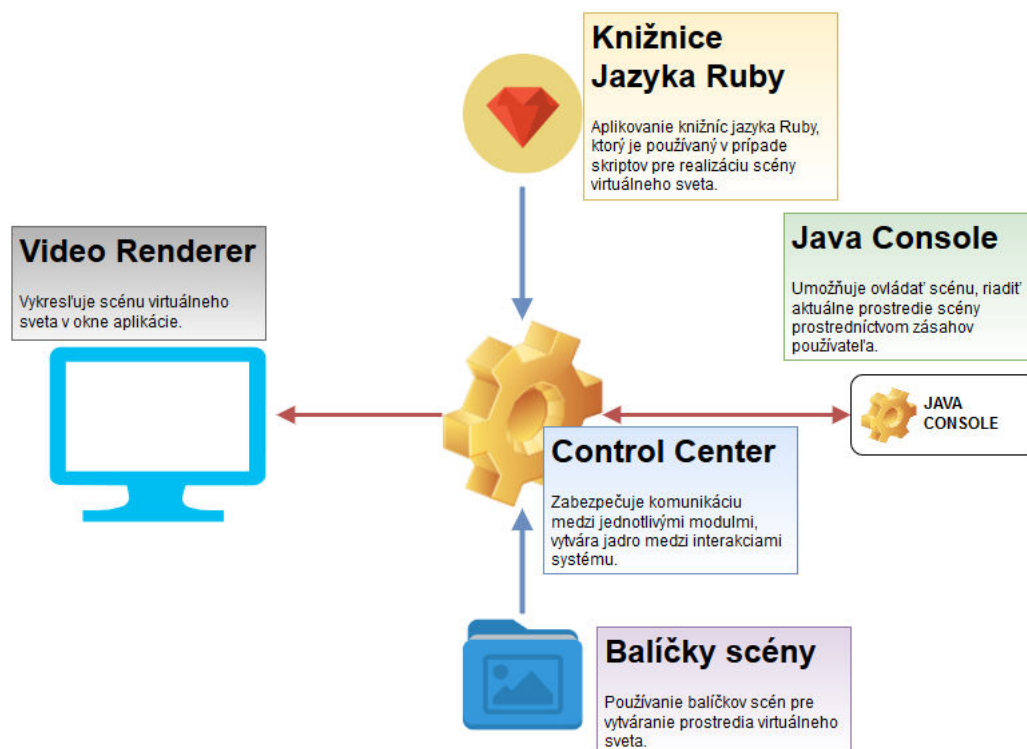
SuperEngine predstavuje softvér pre vytváranie interaktívnej scény v prostredí virtuálnej jaskyne, umožňujúci rozšírenie funkcionalít v trojrozmernom prostredí prostredníctvom skriptovacieho jazyka Ruby [Obr.3]. Systém pozostáva z troch základných modulov so sprostredkovanou komunikáciou pomocou lokálnej siete, radené v nasledujúcom poradí :

- ControlCenter,
- Java Console,
- Video Renderer Win32.

Každý modul je zodpovedný za správny chod systému SuperEngine, ktorý realizuje virtuálny svet z riadiaceho a zobrazovacieho pohľadu virtuálnej scény. ControlCenter reprezentuje jadro systému s vykonávaním základnej interakcie medzi riadiacim prvkom Java Console a vykresľovacím modulom Video Renderer Win32. Pomocou SuperEngine skriptov

implementovaných s využitím dostupných metód SuperEngine v jazyku Ruby umožňuje vytvárať prepojenia medzi scénou a množinou obsiahnutých virtuálnych objektov alebo dát smerujúcich od používateľa prostredníctvom vstupov. Základom modulu ControlCenter je zabezpečenie stability systému a prenosu informácií obsiahnutých v skripte scény alebo prostredníctvom vstupov zasielaných z vytvoreného používateľského rozhrania v module Java Console. Java Console reprezentuje ovládací modul zabezpečujúci spúšťanie virtuálneho sveta s panelom pre riadenie každého renderovacieho počítača, zapínania a vypínania systému virtuálnej jaskyne, možností nastavenia hodnôt parametrizácie zobrazenia scény podľa potrieb používateľa, záznamom chodu nasadeného a spusteného skriptu scény. Rovnako obsahuje panel zobrazujúci variabilné používateľské rozhrania, ktoré je možné implementovať podľa vlastných potrieb. Tlačidlá používateľského rozhrania je možné viazať na jednotlivé časti scény alebo objekty, prípadne vytvárať polia textových znakov, grafické prvky zobrazujúce stav číselných premenných alebo karty s rôznym obsahom informácií. Prostredníctvom Java Console je možné upravené koreňovú štruktúru balíčkov scén s realizáciou pridania balíčka novej scény alebo vymazania niektorého z balíčkov scén. Charakteristické je ovládanie spustenia, pozastavenia alebo ukončenia vykonávania scény virtuálneho prostredia. Z pohľadu parametrizácie scény dovoľuje používateľovi priamo zasahovať v prostredí spustenej scény a vykonávať zmeny parametrizácie hodnôt kamery, zvuku, vzdialeností vykreslenia a úprav pre nastavenia stereoskopického zobrazenia obrazu. Okrem možností riadenia scény dovoľuje používateľovi spravovať systém z pohľadu spustenia, vypnutia alebo reštartovania jednotlivých počítačov v klastri, prípadne vykonávania rovnakých pokynov pre ovládanie Video Renderer Win32. Video Renderer Win32 predstavuje modul prenášajúci štruktúru scény do virtuálneho sveta, zobrazeného renderovaným oknom. Prostredníctvom interakcie medzi Java Console a ControlCenter zobrazuje aktuálnu scénu s obsahom objektov, ktoré sú umiestnené v balíčku scény. V prípade systému virtuálnej jaskyne, s realizáciou sieťového vykresľovania obrazu, je modul Video Renderer Win32 spustený v klastrovej štruktúre na každom výpočtovom systéme zvlášť. Každý počítač v klastri riadenia LIRKIS CAVE obsahuje definovaný počet spúšťaných inštancií okien podľa počtu riadených obrazoviek. Na základe správneho chodu systému sú jednotlivé moduly spúšťané samostatne v rovnakom poradí, v dôsledku ktorého je odosielanie dát medzi jednotlivými modulmi korektné, bez vzniku latencií:

1. Java Console
2. Control Center
3. Video Renderer Win32



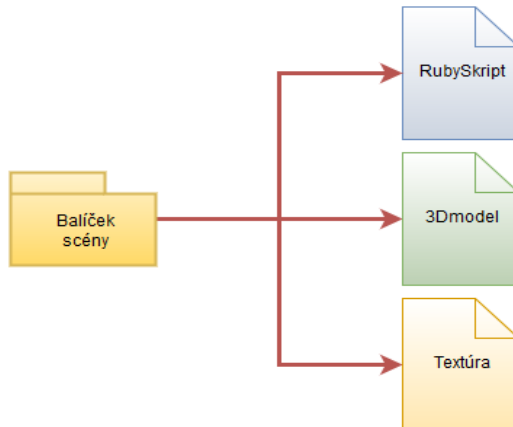
Obr. 3 Zloženie systému SuperEngine.

2.2.2. Vizualizačné jadro OpenSG

Podpora výkonných vizualizačných jadier závisí od druhu systému a zložitosti, pre ktorú je určený. Výhodou OpenSG [8] je podpora rôznych formátov trojrozmerných modelov alebo scén, prípadne obrazových formátov aplikovaných vo virtuálnom prostredí, viazaných na objekty formou textúr alebo realizovaných v prípade vytvárania interaktívneho pozadia (skybox). Jadro OSG poskytuje v prípade využívania softvéru SuperEngine postačujúcu množinu funkcií pre operovanie s jednotlivými prvkami scény s možnosťou nezávislého ovládania objektov alebo jednoduchej aplikácie funkčných vzťahov medzi objektmi. Dôležitým faktorom pre aplikáciu OSG je výkon, ktorého účinky sa prejavujú pri vyššej záťaži spôsobenej detailnou scénou s vysokým počtom polygónov obsiahnutých v trojrozmerných objektoch. Z pohľadu riadenia scény a zabezpečenia interakcií na podnety je OSG účinný, vykonávanie dynamiky scény nepredstavuje vysoké latencie [9]. Absentujúce sú zložky tvorby hernej logiky alebo vytvárania fyzikálnych väzieb pre aplikovanie štandardnej fyziky s využívaním dynamiky scény. Zložitejšie úpravy prostredia v prípade generovania dynamických vlastností reprezentovaných zrýchlením, generovaním správania scény, viazaných na vznik alebo zánik scénických komponentov, prípadne deformáciou objektov sú náročné alebo nemožné. Z pohľadu výkonnostnej stránky nie sú žiadne z predošlých funkcionalít náročné, podstatná je absencia akejkoľvek nadstavby, ktorá by umožnila aplikovanie niektorej funkcionality bez nutnosti opakovanej implementácie pri vývoji prostredia scény.

2.2.3. Importovanie a exportovanie scén

SuperEngine disponuje balíčkami scén s obsahom modelov scénických prvkov, skriptov pre jednotlivé interakcie a funkcionality, zvukových stôp, obrazových formátov aplikovaných textúr objektov [Obr.4]. Štruktúra balíčka scény je funkčná v prípade ak skript scény neobsahuje nekorektné údaje a príkazy, zodpovedá obsahu štruktúry balíčka a nedefinuje neexistujúce objekty, prípadne zariadenia pre ovládanie trojrozmerného prostredia. Importovanie každej scény je jednoduché a intuitívne v prípade ak používateľ nadobúda skúsenosti s prácou v modelovacích nástrojoch a rovnako ovláda skriptovací jazyk Ruby. Každá scéna je jednoducho aplikovateľná, meniteľná prostredníctvom manuálnej editácie balíčka, zámény súborov a opätovným vyvolaním v prostredí konzoly softvéru SuperEngine. Jednou z komplikácií softvéru je absencia možností priamej editácie scény interným nástrojom, ktorý by umožňoval interaktívne rozmiestniť prvky scény s možnosťou exportovania usporiadanej scény priamo do balíčka. Neprítomnosť editácie skriptov scény núti používateľa vykonávať úpravy v externých editoroch, bez okamžitého náhľadu zmeny vlastností scény v momente úpravy. Akceptovateľná je neprítomnosť modelovacieho nástroja, ktorý používateľ vyberá podľa vlastných skúseností, s výsledným importovaním modelov prostredníctvom trojrozmerných formátov *.3ds, *.obj, *.vrml, *.bin podporovaných softvérom.

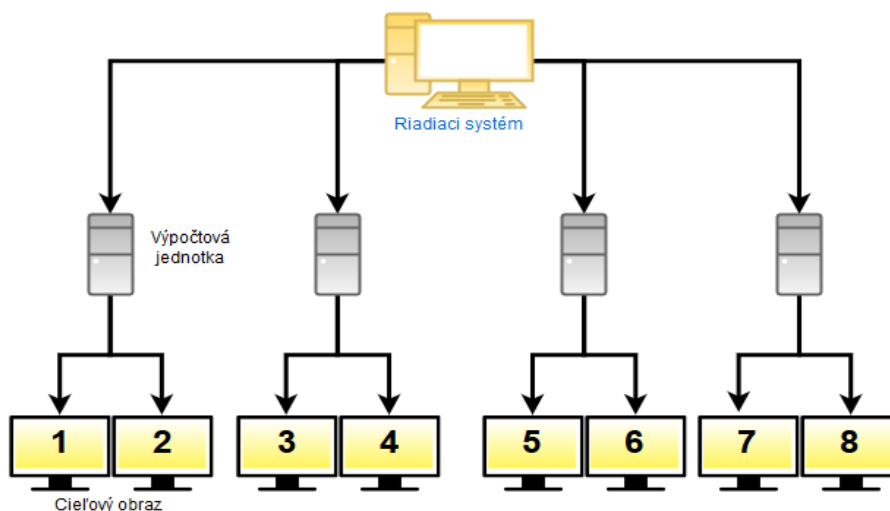


Obr. 4 Štruktúra balíčka scény.

2.2.4. Renderovanie scén

Dôležitou funkciou každého grafického nástroja alebo jadra je vykresľovanie scény s možnosťou konfigurácie z pohľadu rozlíšenia, zložitosti a viditeľnosti vykreslenej scény, parametrizácie uhla perspektívy zobrazenia, riadenia častí grafického reťazca (shader). Rovnako podstatný je systém realizujúci vykresľovanie v prípade zdieľania obrazu prostredníctvom viacerých výstupných zariadení. V prípade systémov virtuálnych jaskýň s prioritne vyššími grafickými nárokmi scény sú aplikované výpočtové klastre, umožňujúce vykresľovanie s delením

scény na niekoľko výstupov prostredníctvom rozdelenia vykresľovacích úloh podľa veľkosti scény, obrazu medzi jednotlivými časťami klastrového systému. Bao [10] pri používaní klastrovej hierarchie potvrdzuje zlepšenie vlastností systému z pohľadu náročnosti objektov scény a zlepšenie výkonu prostredníctvom reakcií systému. V prípade riadenia systému v klastrovej štruktúre je možné rozdeliť časti scény podľa druhu tvaru jaskyne a tým definovať konfiguráciu ostatných systémov zabezpečujúcich vykresľovanie obrazu v jednotlivých častiach. Dôležité je uviesť spôsob delenia scény v prípade riešenia systému v klastrovej štruktúre. Hlavnú časť vykreslenia scény systému virtuálnej jaskyne reprezentuje riadiaci systém, ktorého úlohou je rozdelenie pohľadov scény pre jednotlivé výpočtové jednotky a poskytovaním údajov o stave scény pre každú výpočtovú jednotku zvlášť. V takomto prípade je riešenie systému virtuálnej jaskyne z logického pohľadu triviálne, bez výrazných požiadaviek kladených na náročnosť a záťaž výpočtového systému. Ďalšia z výhod predstavuje poskytovanie globálnych údajov scény pre každú výpočtovú jednotku. Globálne údaje o aktuálnom pohľade kamery, rozsahu scény, sú obsiahnuté v riadiacom systéme [Obr.5], poskytované pre každú výpočtovú jednotku v klastri. Výpočtové jednotky sú schopné prevziať globálny údaj s aplikáciou pre vlastné lokálne prostredie v trojrozmernej scéne, ktoré podľa preddefinovanej konfigurácie upravujú a zobrazujú v systéme virtuálnej jaskyne. Charakteristický globálny údaj môže reprezentovať pozícia alebo orientácia kamery v scéne. Každá výpočtová jednotka obsahuje vlastný konfiguračný údaj pozostávajúci s individuálnou orientáciou kamery a údajmi pre rozlíšenie, hĺbku, stereoskopické zobrazenie. V prípade poskytnutia globálneho údaju je každá výpočtová jednotka autonómna a vykresľuje len definovanú oblasť, pričom nie je nutné na každej výpočtovej jednotke vykresľovať opakovane celú scénu. Takto je možné deliť jednotlivé časti scény a zabezpečiť ľubovoľný počet výstupov pre zobrazovacie zariadenia.



Obr. 5 Rozdelenie úloh vykreslenia obrazu.

2.2.5. Statické objekty scény

Objekty nevykonávajúce zmenu tvaru, veľkosti alebo pozície v akomkoľvek časovom úseku simulácie scény sú považované za statické objekty virtuálneho sveta. Každý statický objekt je reprezentovaný vlastnosťami definujúcimi jeho charakter, veľkosť, škálovanie, materiál, prípadne textúru. Jerald [11] považuje využívanie objektov statickej scény za dôležité v prípade tvorby terénu alebo povrchu s rozsiahlou plochou, prípadne vytáranie statického podložia pre aplikáciu dynamiky modelov, ktoré sú na statickú scénu viazané. Medzi základné statické objekty sú radené rôzne primitívne modely, prípadne množina bodov alebo vzťažný uzol. Rovnako tieto prvky môžu byť dynamické, v prípade aplikácie rôznych funkcií pre zmenu vonkajších vlastností objektov. Vonkajšie vlastnosti statických objektov nie sú menné, reprezentujú statickú štruktúru, zachovávajúcu akékoľvek vlastnosti v realizácii simulácie. Prípustný je vnútorný stav objektov, napríklad počítadla kolízií, zmena fyzikálnej vlastnosti objektu (napríklad teplota), ktoré môžu v čase nadobúdať menný charakter. V prípade SuperEngine sú statické objekty opísateľné rôznym spôsobom s viazaním na premenné obsiahnuté v skripte scény alebo ich statickými nemennými hodnotami akými sú poloha, otočenie, škálovanie. Dôvodom používania statických objektov je rovnako poskytovanie vzťažných bodov scény, bez zmeny rozmerov alebo vonkajších vlastností.

2.2.6. Dynamické objekty scény

Každý interaktívny trojrozmerný systém poskytuje dynamické prostredie objektov, disponuje dynamickými reakciami prostredia na podnet používateľa prostredníctvom vykonávania zmien v scéne. Každý dynamický objekt vykonáva činnosť v čase a priestore so zmenou vonkajších vlastností. Medzi základné dynamické vlastnosti trojrozmerného objektu sú radené posuny objektu v trojrozmernom súradnicovom systéme, rotácia, škálovanie. Podľa Sobotu [12] využívanie dynamickej scény prináša reálnejší vnem pri interakciách vznikajúcich medzi používateľom a systémom. Dôležité je zameranie sa na kombináciu efektov používaných v rámci dynamickej scény, reprezentujúcich rôzne zmeny vonkajších vlastností objektov. V prípade prostredia SuperEngine je možné vykonávať rôzne funkcionality dynamickej scény s kombinovaním transformácií, deformácií alebo nárazov objektov. Objekt dynamickej scény je rovnako nositeľom vnútorných vlastností, v porovnaní s objektom statickej scény, ktoré môže v priebehu času simulácie meniť. Rozdiel je zahrnutý v aplikácii vnútorných vlastností s možnosťou ovplyvnenia vonkajších vlastností a opačne. Príkladné je využívanie objektu s vnútornou vlastnosťou teploty a viazaním vonkajších vlastností podľa aktuálnej hodnoty teploty objektu. V prípade nižších hodnôt objekt vykazuje mierny pohyb, obsahuje jednoduchú textúru studenej farby. Zvýšením teploty bude objekt vykonávať rýchlejšie dynamické pohyby a meniť farbu spojito

do teplejších odtieňov. Uvedený príklad charakterizuje často používané princípy v trojrozmerných virtuálnych dynamických scénach, v prípade ktorých je možné vykonávať zložitejšie efekty a postupnosti vytvárania dynamického prvku.

2.2.7. Svetelné prvky

Dôležitý prvok trojrozmernej scény prostredia predstavuje svetelné objekty používané pri vytváraní reálnejšieho vnemu na používateľa, obohacujúce scénu o efektové riešenie s prirodzenejším vzhľadom a umožňujúce zvýrazniť miesta scény používané v popredí. V systémoch virtuálnej reality je scéna svetelných prvkov rozdelená na globálny a lokálny priestor dopadu svetelných lúčov. Globálna scéna reprezentuje celý virtuálny svet vrátane všetkých obsiahnutých objektov scény, nakoľko lokálna scéna zobrazuje pod-objekty globálnej scény s možnosťou výberu kombinácie osvetlenia objektov. Globálne svetelné prvky sú v prípade aplikácie na trojrozmernú scénu využívané v prípadoch generovania slnečného žiarenia využívaním ostrejších lúčov s nízkym, prípadne zanedbaným rozptylom, alebo svetelného žiarenia prostredníctvom jemných prechodov lúčov s miernym rozptylom v každej časti scény [13]. Opačný prípad zaznamenáva scéna s využívaním lokálnych svetelných prvkov, ktorých používanie môže predstavovať náročnejšiu aplikáciu so zložitou závisiacou od tvaru objektov obsiahnutých v trojrozmernej scéne virtuálneho prostredia. V prípade globálnej a rovnako lokálnej scény je množina svetelných prvkov rozdelená na dve základné skupiny obsahujúce svetelné prvky s lúčom rozptylu a svetelné prvky s ostrým lúčom. Rôznorodé môže byť nastavenie svetelného prvku, zaoblenie okrajov lúča, zaostrenie alebo vytváranie rozptylového efektu.

2.2.8. Zvukové prvky

Objekty virtuálneho sveta môžu nadobúdať rôzne vlastnosti v prípade, ak nástroj pre tvorbu virtuálnej scény alebo využívanie skriptovacieho jazyka v kombinácii dovoľujú používanie rozšírenia efektových prvkov, napríklad aplikáciu zvukových efektov. V prípade použiteľnosti zvukových efektov je rovnako deliteľná scéna z globálneho alebo lokálneho pohľadu v rámci sledovania prostredia alebo objektu. Qvortrup [14] zdôrazňuje dôležitosť viazania zvuku podľa spôsobu priebehu interakcie v prostredí virtuálneho sveta. Z globálneho pohľadu je možné nasadenie zvuku prehrávaného v slučkách, alebo v prípade výskytu udalostí vykonávanie zmeny vo všetkých častiach scény. Príkladné používanie zvuku prostredia je opísateľné zmenami počasia, pri ktorých celá scéna nadobúda zmenu zvukových vlastností, napríklad padajúceho dažďa. V aplikáciách lokálnych efektov je možné zaviesť zvukový prvok v akomkoľvek bode scény s prípadným viazaním na konkrétny objekt alebo časť scény. Dynamický objekt môže byť nositeľom zvuku so sprevádzaním počas vykonávania transformácií alebo v stave bez ich aplikácie.

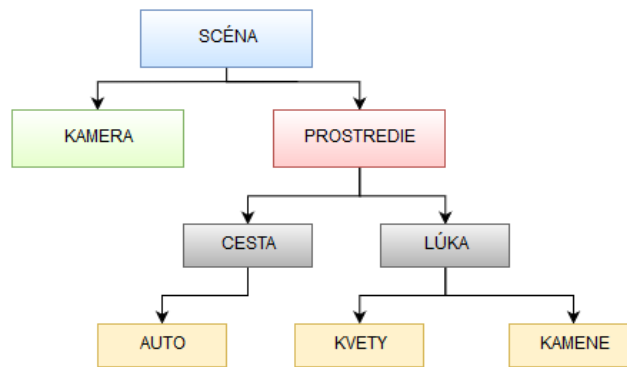
Príkladným opisom je realizácia vozidla so zapnutým alebo vypnutým motorom, pri pohybe alebo státi, kedy je možné kombinovať tieto vlastnosti so zvukom viazaným na objekt auta. SuperEngine umožňuje realizáciu s kombináciou statických a dynamických zvukových prvkov, viazanie zvuku formou globálneho prostredia alebo jednotlivo pre každý objekt virtuálneho sveta osobitne.

2.3. Praktické dôsledky využívania systému

Systém virtuálnej jaskyne s nasadením softvéru vytvoreného spoločnosťou Slovakia SuperComputers umožňuje vytvárať rôzne druhy prostredí virtuálneho sveta s možnosťou realizácie dynamického správania a simulácie scény. Prostredníctvom využívania kompozície scény poskytuje vytváranie grafu obsiahnutých objektov, samostatne alebo viazane formou vzťažných bodov vytvárajúcich vetvy grafu s definovaním ďalších častí trojrozmerných modelov virtuálneho prostredia. Z pohľadu simulačných vlastností disponuje ovládacími prvkami pre plynulé ovládanie (snímanie polohy používateľa v systéme), prípadne ovládania využívaním výskytu udalostí v systéme alebo manuálnym zásahom používateľa s využitím implementovaného hardvérového prvku. V prípade simulácie je potrebné doceliť dostatočnú funkcionality a zložitosti scény neprekračujúcu výkon poskytovaný výpočtovým systémom. Limity systému môžu spôsobovať časté latencie alebo nepresnosti s dopadom na priebeh interakcie medzi používateľom a výpočtovým systémom, riadiacim reakcie scény virtuálneho sveta. Správnym konfigurovaním scény pre jej rozdelenie v systéme klastra, znalosťami o dostupnom výkone systému a rozdelením úloh CAVE systému v správnom poradí je možné doceliť finálny tvar aplikácie bez nežiadaného vykonávania.

2.3.1. Kompozícia scény

Každá scéna reprezentovaná softvérom SuperEngine pozostáva zo základného grafu scény schopného realizovať väzby medzi jednotlivými objektmi, logicky rozdeľovať scénu na komponenty a vzťažné uzly. Základom kompozície scény prostredníctvom skriptovacieho nástroja v jazyku Ruby je možné definovať viacero vzťažných ulov, ktoré sú viazané na počiatkový bod trojrozmernej súradnicovej sústavy prostredia [Obr.6]. Objekty virtuálneho prostredia reprezentujú výsledné zobrazenie grafu scény, ktorý zodpovedá vlastnostiam virtuálneho sveta zobrazeného v CAVE systéme. V prípade softvéru SuperEngine sú k dispozícii objekty kamery, trojrozmerné modely objektov, zvukové prvky, uzly s vlastnosťou viazania prvkov scény. Delenie grafu scény závisí predovšetkým od funkčného používania a dekompozície súčastí jednotlivých objektov, prípadne prostredia, pre ktoré bude výsledný systém interakcie poskytovať.



Obr. 6 Názorná ukážka grafu scény.

V prípade vyjadrenia grafu dynamickej scény je možné upravovať jednotlivé uzly podľa spôsobu realizácie interakcie, rovnako z pohľadu kompozície objektu [15]. Každú scénu je možné opísať grafom so zohľadnením jednotlivých väzieb, spájajúcich jednotlivé objekty s vytvorením finálneho prvku. Graf scény poskytuje dokumentáciu väzieb akéhokoľvek trojrozmerného prostredia s nasadením objektov ako komponentov a pod komponentov cieľového modelu.

2.3.2. Simulácia scény

Vytváranie dynamických scén prostredníctvom modelovacieho a skriptovacieho nástroja je nutné testovať pre prípady overenia správnosti vykonaných transformácií alebo nepresností pri importovaní niektorého z objektov. Systém SuperEngine absentuje interným nástrojom pre implementáciu skriptov scén s automatickou kontrolou správnosti zápisu metód. Nevýhodou je narastajúca chybovosť pri zvyšovaní náročnosti scény, ktorú je potrebné postupne overovať spúšťaním simulácie scény v systéme. Simuláciu je možné ovládať kontinuálne na základe implementovaného hardvérového ovládacieho prvku alebo predurčiť jej realizáciu prostredníctvom postupnosti javov prebiehajúcich vo virtuálnom prostredí. Využívaním prostredia SuperEngine je možné simuláciu zastaviť v prípade potreby a opätovne pokračovať od začiatku alebo momentu, v ktorom bola simulácia pozastavená. Výhoda aplikácie Ruby skriptov prináša rozšírenie ponuky simulačného prostredia pre obohatenie funkcionality formou dotykových tlačidiel, prípadne senzorov, prepínačov alebo interaktívnych okien so zobrazovaním aktuálneho stavu objektov na pozadí simulovanej scény [16]. V prípade používania rôznych druhov scén, je možné vytvárať rôzne kombinácie funkcionalít viazaných na vstupy a výstupy systému. Z pohľadu zovšeobecnenia systému je výhodou vytváranie akýchkoľvek simulačných prostredí bez výrazného obmedzenia používateľa z pohľadu priestoru alebo tvaru systému jaskyne. Samotná simulácia scény umožňuje vykonávanie rôznych scén s ohraničením spôsobeným limitmi systému.

2.3.3. Limity systému

Úlohou každého systému je splniteľnosť požiadaviek, ktoré sú kladené pri riešení problému. V prípade systému od Slovakia Super Computers je možnosť realizácie požiadaviek širokospektrálna, predovšetkým pre využitie v odvetviach využívajúcich simuláciu v trojrozmerných prostrediach. Systém umožňuje kombinovať jednotlivé funkcionality s priradenou dynamikou scény, aplikovať vykonávanie funkcionalít pre rozšírenie používateľského rozhrania, implementáciu hardvérových ovládačov a prvkov, zavádzanie moderných systémov z pohľadu snímania polohy používateľa. Druhá stránka reprezentuje pohľad na výkon systému pri vytváraní scén s vysokým počtom polygónov objektov, prípadne zložitými štruktúrami vo veľkom počte. Podľa Forsytha [17] výsledný efekt v takomto prípade môže zabrániť vhodnému vykresleniu, respektíve preťažovať systém s požiadavkami pre opakované vykresľovanie častí, komponentov scény. Možnosťou dosahovania vykresľovania scén so zložitou štruktúrou bez výrazného preťažovania jadra systému je aplikácia metodík využívajúcich zjednodušenie štruktúry scény, bez dopadu na jej kvalitu a členitosť. Komplexný problém pri realizácii vykresľovania zložitých scén predstavuje tieňovanie objektov, ktoré v prípade vysokých požiadaviek na výkon výpočtových systémov môže spôsobovať časté latencie alebo nesynchronne zobrazovanie jednotlivých obrazov s časovým rozdielom postrehnuteľným voľným okom. Takýto prípad poukazuje na latencie spôsobené používaním viacerých svetelných prvkov scény, používaných spoločne v rovnakom čase, alebo nesprávnou kompozíciou scény z pohľadu používania objektov s náročnou a jednoduchou štruktúrou. V prípade výrazných latencií je nutné objekty scény porovnať z pohľadu zložitosti a ustáliť ich počet v norme pre zabránenie spomalenia systému. Norma nie je viazaná staticky, pre každú scénu a systém je individuálna podľa dostupného výkonu. V základnom pohľade predstavuje hranicu zložitosti scény medzi vykonávaním v stave s latenciami a bez latencií. Vytváraním scény je potrebné zohľadniť maximálny a optimálny výkon systému s prostriedkami dostupnými pre vizualizáciu scény. Systém SuperEngine obsahuje dostupné prostriedky pre riadenie scény na úrovni bežných vizualizačných jadier. V porovnaní s prvkami pri vytváraní efektov scény, svetelných alebo časticových je systém výrazne obmedzený, aplikovanie prvkov efektov je nutné riešiť prostriedkami s možnosťou použitia externých modelovacích nástrojov a metodík pre zachovanie efektu z pohľadu iného riešenia. Celkovo je systém SuperEngine striktno obmedzený z pohľadu výkonu pri vykresľovaní zložitých štruktúr alebo textúr v porovnaní s ostatnými dostupnými vizualizačnými jadrami.

2.3.4. Interakcie s používateľom

Využívanie virtuálnych jaskýň z pohľadu interakcií je obohatené o rôzne hardvérové prvky, ktoré dopĺňajú funkcie systému a dovoľujú používateľovi zasahovať do prostredia virtuálneho sveta. Rovnako umožňujú vytváranie interakcií viacerých používateľov reálnom čase v rámci jednej alebo viacerých virtuálnych jaskýň. Základná konfigurácia jaskyne používanej v LIRKIS CAVE disponuje snímaním pohybov hlavy jednej osoby nasadenej priamo do prostredia, pričom ostatné osoby plnia úlohu pozorovateľov virtuálneho priestoru. V skutočnosti sledujú obraz, ktorým reaguje systém na pozíciu hlavy používateľa snímaného zariadením OptiTrack. Každý z používateľov je v reálnom čase prítomný, pričom hlavný používateľ vykonáva interakcie so systémom, ktorý prispôsobuje vykreslenie obrazu prostredia a mení uhol pohľadu v scéne. V prípade aplikácie scén rôzneho druhu je systém variabilný, umožňuje vytvárať rôzne pohľady, definovať správanie trojrozmerných objektov a prenášať funkcie medzi jednotlivými komponentmi systému v logických väzbách. V prípade hardvérovej konfigurácie systém disponuje rozlíšením 20xK, distribuovaným s využitím 6 pracovných staníc podieľajúcich sa na vykresľovaní. Aplikovanie scén je v prostredí jednoduché, modulárne, a dovoľuje kombinovať rôzne prvky s využitím skriptov jazyka Ruby. Systém jaskyne umožňuje dopĺňanie rôznych rozšírení v rámci interakcií v trojrozmernom virtuálnom prostredí. Z pohľadu používania hardvérových zariadení je realizovateľné využívanie vstupov prostredníctvom profesionálnych ovládacích prvkov. Jeden z charakteristických prvkov pre pohyb kamery a používateľa na scéne je 3D mouse [18], reprezentujúci jednoduché zariadenie ovládané jednou rukou používateľa. Ďalšie zariadenie predstavuje Intersense Wand [19], podporujúci sledovanie pohybu ovládacích zariadení v ruke používateľa. Jedným z najaktuálnejších zariadení a inovatívnych technológií je používanie systému sledovania polohy používateľa prostredníctvom sledovania bodov s využívaním infračervených kamier OptiTrack [20]. Rovnako je možné doplnenie na báze využívania softvérových modulov pre rozšírenie podpory vstupných zariadení. Každé hardvérové zariadenie je variabilné pre prispôbenie k riadeniu rôznej funkcionality scény, s možnosťou prispôbovať správanie scény podľa prúdu vstupných údajov alebo výskytu udalostí prostredníctvom hardvérového zariadenia so zachytávaním pohybov, stláčania tlačidiel.

2.4. Nasadenie virtuálnych jaskýň

Kapitola opisuje základné aplikácie virtuálnych jaskýň v prostrediach simulačných systémov, edukácie a zábavy z pohľadov používateľa, výhod a nevýhod vyskytujúcich pre jednotlivé nasadenia. Prostredníctvom kapitoly simulačných systémov porovnáva nasadenie virtuálnych scén a hardvérových prvkov používaných v činnosti simulácií situácií alebo strojov vhodných pre realizáciu prostredníctvom vizualizácie prostredia v priestore virtuálnej jaskyne.

Z pohľadu edukačnej činnosti poukazuje na využívanie virtuálnych prostredí zameraných na vykonávanie činností v rámci odborov univerzít na území Slovenskej Republiky s porovnávaním výhod finančných nárokov pri využívaní virtuálnych prostredí oproti vyšším finančným nákladom vynaloženým pri realizácii skutočnej fyzickej scény alebo prostredia. v závere kapitola opisuje nasadenie systému virtuálnej jaskyne z pohľadu zábavy a voľného času s konštatovaním výhod pre zavádzanie moderných technológií a dosahovaní reálnejšieho vnemu používateľa. Argumentáciou vyššej fyzickej bezpečnosti potvrdzuje zobrazovanie prostredí, ktorých skutočná realizácia môže byť životu nebezpečná alebo nevhodná v prípade adrenalínových športov, životu nebezpečných situácií.

2.4.1. Simulačné systémy

Simulačné jednotky využívajú systém virtuálnej jaskyne s možnosťou realizovania širokého spektra scén alebo prostredí. Využívanie simulačných zariadení prostredníctvom virtuálnych jaskýň je častým riešením s možnosťou variabilne pracovať s rôznymi parametrami ovplyvňujúcimi simuláciu scény podľa druhu virtuálneho prostredia. Niektoré jaskyne sú definované špeciálnym tvarom s obsahom rôznych hardvérových systémov, napríklad simulátory strojov. Výhodou využívania typizovaného tvaru virtuálnej jaskyne reprezentovaného kockou, kruhom alebo polkruhom je obmena aplikácie rôznych scén, pre ktoré je systém variabilný. Výhoda prispôsobivosti virtuálnej jaskyne dovoľuje širokospektrálnu aplikáciu s nižšími nákladmi realizovateľnosti oproti špeciálne stavaným systémom [21]. Bežné simulačné systémy využívajú jeden druh simulácie, pre ktorý sú aplikované z hľadiska používateľa alebo situácie, prípadne z pohľadu potrebného pri riešení vybranej problematiky alebo situácie. Systémy podporujúce širšie spektrum aplikácie umožňujú obmieňať rôzne prostredia bez výrazných potrieb zmeny hardvérovej časti, v druhom prípade nedovoľujú vykonávať bližšiu špecializáciu pre riešenie vybraného problému. Každý druh vykonávanej simulácie poskytuje používateľovi minimálne základný pohľad pri hľadaní riešenia problému, pre ktorý je simulačné prostredie navrhnuté. Príkladný pohľad je možné uviesť pri realizácii ovládání stroja, prostredníctvom špeciálne odbaveného systému virtuálnej jaskyne [Obr.7]. Používateľ aplikuje v prípade simulácie reálne hardvérové prvky kopírujúce reálny systém s možnosťou vstupovať do simulácie s presne charakteristickým využívaním. Vhodné je využívanie simulačných zariadení pre ovládanie žeriavov, lietadiel alebo iných strojov. Jaskyňa nedisponujúca špeciálnym hardvérovým zariadením kopírujúcim skutočný systém môže poskytovať podobné hardvérové zariadenia formou prispôsobenia ovládacích prvkov klávesnice, myši, prípadne riadenie upravené pre Joystick. V prípade realizácie zložitých simulačných scén je možné dosahovanie efektov vytvárajúcich presvedčivosť používateľa o jeho umiestnení v simulovanom prostredí. Efekty môžu byť

reprezentované formou hardvérových prvkov, napríklad hydraulických plošín alebo vibračných spätných väzieb zasahujúcich do prostredia používateľa, alebo softvérovým spôsobom dotvárajúcim dynamické efekty, prípadne zvukové efekty viazané k trojrozmerným objektom v scéne simulácie.



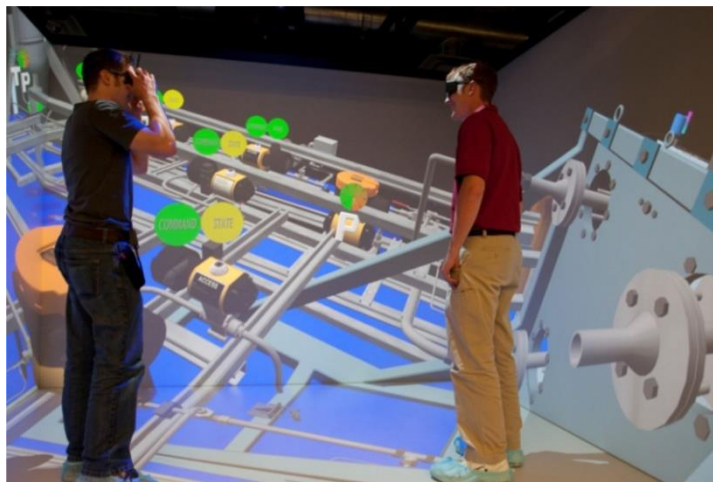
Obr.7 Simulácia stavebného vozidla v prostredí virtuálnej jaskyne. [22]

Používateľ v takomto prípade využíva simuláciu na úrovni vnemu blízkeho skutočnosti, aplikuje ale ovládacie zariadenia s nižšou úrovňou riadenia oproti reálnemu zariadeniu umiestnenom v ovládanom stroji. V porovnaní so softvérovým odbavením jaskyňa vykonávajúca jeden typ simulácie obsahuje špeciálny softvér a vizualizačné jadro, spĺňajúce úroveň náročnosti a realizovateľnosti zvolenej simulácie. v opačnom prípade je softvérové zabezpečenie virtuálnej jaskyne pre vykonávanie viacerých druhov simulácií podporované vizualizačným jadrom a vhodnou implementáciou softvérových modulov pre realizáciu požadovanej scény alebo prostredia s možnosťou variabilnej zmeny.

2.4.2. Edukačné prostredia

Systémy virtuálnej reality aplikované pre edukačné účely s využitím virtuálnych jaskýň poskytujú široké spektrum možností v rôznych odvetviach edukácie. Podľa Sterna [23] trojrozmerné prostredie otvára uplatnenie pre interakciu medzi používateľom a trojrozmerným svetom, ktorý umožňuje vnímať riešený problém presnejšie, definovanejšie. Edukácia prostredníctvom virtuálnych jaskýň je realizovateľná v akomkoľvek odbore s rôznym zameraním. Príkladné využitie na území Slovenskej Republiky je obsiahnuté vo výskumnom centre virtuálnej jaskyne Technickej Univerzity vo Zvolene. Systém aplikuje edukáciu pre študentov v odbore lesníctva prostredníctvom práce s nástrojmi pre označenie rôznych porastov s evidenciou v databáze, zobrazenie lokality lesa na vybranom území Slovenskej Republiky a riešenie odhadu doby rastu drevín. Edukačný systém aplikovaný na území Technickej Univerzity v Košiciach

zahŕňa podporu výučby predmetov so zameraním v smere počítačovej grafiky s využívaním hardvérových prvkov pre zvukový vnem, transformáciu objektov na scéne alebo sledovaním pohybovej aktivity používateľa vo virtuálnej jaskyni. Každý z uvedených systémov univerzít je fyzicky rozdielny, softvérovo využívajú rovnakú platformu s použitím softvéru od spoločnosti Slovakia Super Computers. Svetové aplikovanie systémov virtuálnych jaskýň pre účely edukácie je aktuálne intenzívne orientované vo výrobe strojárenských zariadení, konštrukcií a architektúre s možnosťou zasahovať do prostredia a interaktívne meniť vlastnosti objektov [Obr.8].



Obr.8 Edukačný systém pre kompozíciu palivových článkov NASA. [24]

Výhodou výučby prostredníctvom virtuálnej reality je disponovanie virtuálnymi objektmi blízskymi reálnym objektom, ktorých finančná realizácia by v súčasnosti predstavovala vysoké finančné nároky alebo v prípade medicínskych zariadení podporuje výučbu s aplikáciou praxe v prostredí virtuálneho sveta [25].

2.4.3. Výskum

Využívanie systémov virtuálnej reality v prospech výskumu je úzko prepojené s edukáciou, ktorá aplikuje poznatky pre rozšírenie podpory edukačných systémov. Aktuálne trendy výskumu sa zameriavajú smerom zobrazovania reálnejších prostredí, generovaním svetelných prvkov svetla, využívaním prvkov umelej inteligencie, prenášaním mračien bodov objektov a štruktúr, aplikovaním fyzikálnych vlastností a zákonov. Každý z uvedených faktorov predstavuje identickú vlastnosť systému s možnosťou aplikácie pre rôzne odvetvia výskumu. Z pohľadu Geisera [26] odvetvie najviac prosperujúce vo výskume prostredníctvom využívania systémov virtuálnej reality a virtuálnych jaskýň predstavuje biomedicínske inžinierstvo využívajúce rôzne štruktúry a odbor medicíny pri vykonávaní operácií s rôznymi časťami štruktúr ľudského tela [Obr.9]. Využívanie skenerov pomocou snímania mračien bodov dovoľuje prenášať rôzne

štruktúry v rámci pevných materiálov, ľudského tela alebo objektov priamo do prostredia virtuálnej reality.



Obr.9 Aplikácia prostredia virtuálnej jaskyne v medicínskom výskume. [27]

Dôležitou témou výskumu je realizácia správania objektov na podnety v závislosti od materiálu vytvárajúceho ich štruktúru. Výhodou v uplatnení výskumu štruktúr je vytváranie rôznych prostredí s rozdielnymi štruktúrami a porovnávaní ich reakcií na podnety s aplikáciou fyzikálnych zákonov, miesto fyzickej realizácie a vysokých finančných nárokov pri reálnom používaní rôznych materiálov. Výskum v prostredí virtuálnej jaskyne je realizovateľný z pohľadu viacerých používateľov v rovnakom čase, podobný pozorovaniu testovanej situácie v reálnom prostredí.

2.4.4. Zábava a voľný čas

Využívanie aplikácií pre zábavu rozširuje zručnosti a reflexy používateľov s tréňovaním vnemu a reakcií na podnety systému. Dôležitým faktorom pri realizácii scény prostredia pre úroveň zábavy je dosahovanie jednoduchosti a primeranej rýchlosti scénických prvkov oproti simulačným technológiám a edukačným systémom. Prostredia reprezentujú pohybové aktivity vrátane adrenalínových scén podnecujúcich používateľa s vytváraním klamlivého vnemu, riešiteľné úlohy z logického pohľadu v prípade kompozície objektu z častí, estetické vnímanie scény prostredníctvom aplikácie virtuálnej galérie, alebo zabudovaním interaktívnych častí formou videí v plochách objektov pre futuristický dojem. Výhodou v riešeníach takýchto prostredí je nulové opotrebovanie trojrozmernej scény oproti reálnym objektom, fyzicky používaným v prostredí reálneho sveta. Využívanie virtuálnej reality prostredníctvom virtuálnej jaskyne alebo virtuálnej helmy je oproti bežnej fyzickej realizácii bezpečnejšie, v prípade vykonávania zábavnej činnosti s možným ohrozením ľudského života. Príkladné aktivity opisujú adrenalínové športy alebo rôzne iné aktivity vyžadujúce dlhodobé skúsenosti používateľa v reálnom svete. Okrem vykonávania nebezpečných aktivít

prostredníctvom zábavy formou virtuálnej reality je možné aplikovať systém z estetického hľadiska prostredníctvom virtuálnych koncertov, divadelných predstavení alebo iných umeleckých podujatí [28]. Aplikácia virtuálneho sveta pre zábavu prináša neustále inovácie senzorických systémov, ktoré sa častejšie zdokonaľujú pre jednoduchšie ale presnejšie používanie v reálnom prostredí.

2.5. Trojrozmerná scéna virtuálnej jaskyne

Základným cieľom virtuálnej jaskyne je docielenie výsledku realizácie trojrozmerného prostredia s obsahom virtuálnej scény. V dnešných aplikáciách je možné scény rozdeľovať podľa rôznych typov nasadenia z druhu pohľadu vykonávaných úloh, dostupných prvkov hardvéru, parametrizácie a spôsobu používaných metodík. Každá scéna je deliteľná na jednotlivé komponenty charakterizujúce scénické prvky vrátane prostredia, kamery, objektov a interakcií, ktoré sa medzi nimi uskutočňujú. Zloženie každej scény je odlišné, reprezentované môže byť rovnako vzťahmi medzi objektmi prostredníctvom uzlov tvoriacich hierarchiu objektov scény. Okrem vytvárania vzťahov medzi objektmi, prípadne viazania jednotlivých častí scény, alebo aplikáciou transformačných uzlov existujú ďalšie prvky, dôležité pre dosahovanie vyššej úrovne reálneho vnemu. Takéto prvky predstavujú množinu svetelných efektov, snahu o vytvorenie reálne pôsobiaceho okolia prostredia, alebo viazania dynamiky scény medzi súvisiacimi prvkami. Spájanie trojrozmerných prvkov do výslednej formy z pohľadu logického súvisu a dosiahnutia konkrétneho vizuálneho stavu môže mať rozhodujúci efekt pre dosiahnutie reálneho vnemu v trojrozmernom prostredí.

2.5.1. Charakteristika scénických prvkov

Virtuálne prostredie v SuperEngine od Slovakia Super Computers charakterizuje trojrozmerná scéna s obsahom objektov reprezentujúcich scénické prvky potrebné pre realizáciu finálnej podoby. Každý scénický prvok môže disponovať odlišnými vlastnosťami z vizuálneho alebo funkčného pohľadu, podľa špecifikovanej úlohy využitia v scéne. Trojrozmernú scénu aplikovanú v systéme virtuálnej jaskyne je potrebné presnejšie definovať podľa požadovaného účelu jej nasadenia na jednotlivé scénické parametre:

- prostredie scény,
- rozsah scény,
- formáty trojrozmerných objektov,
- typ používanej kamery,
- svetelné podmienky,
- špeciálne efekty.

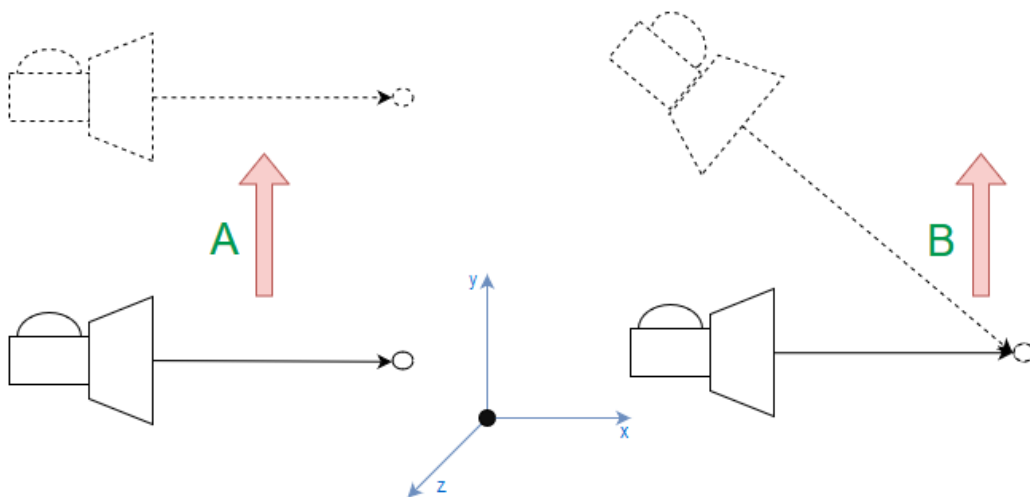
Prostredie scény predstavuje hlavný parameter scény, od ktorej sa odvíjajú ďalšie postupy pri jej budovaní. Za príkladné prostredia scény je možno považovať rôzne reálne prostredia skutočného sveta podľa požadovanej úlohy (napríklad parky, budovy, miestnosti), ktorá má byť v prostredí realizovateľná. Prostredie scény musí vplyvať primerane z pohľadu vybranej realizovanej aktivity, bez negatívneho vplyvu na používateľa s prínosom reálnejších pocitov pri vykonávaní interakcií medzi používateľom a scénou. Ďalší scénický parameter opisuje rozsah aplikovanej scény s vymedzením krajných bodov, hraníc obsiahnutých v prostredí scény. Podľa Cornelisa [29] je možné každú scénu definovať vyčlenením konkrétnych prvkov, alebo hraníc, prostredníctvom ktorých je možné presnejšie určiť rozsah vykonávanej úlohy. V prípade realizácie scén aplikovaných s využitím miestností sú hranice rozsahu definované veľkosťou miestnosti a jej tvarom. V prípade vonkajších scénických aplikácií je rozsah definovaný rôznymi spôsobmi pre vyčlenenie hraníc s využitím rozsahu plochy podlažia, vytvorením hraničnej línie, alebo využívaním maximálnej vzdialenosti pohľadu kamery. Voľba formátov trojrozmerných objektov závisí od zložitosti požadovanej scény a funkcionalít v nej obsiahnutých. V prípade realizácie jednoduchších scénických transformácií a objektov je vhodné aplikovať formát *.3ds, ktorý obsahuje jednoduchú a štandardizovanú štruktúru s nižšími nárokmi pre požadované výpočty v rámci triangulácie plôch, svetelných vlastností, materiálov a textúr, údajov o pozícii kamery a umiestnení objektu v importovanom súradnicovom systéme. Zložitejšie typy štruktúr trojrozmerných objektov reprezentuje z geometrického hľadiska formát *.obj, orientovaný formou spresnenia geometrických vlastností objektu za účelom detailnejšieho vnímania vertexov (bodov spájajúcich hrany). Možné je uvažovať o ďalších rôznych formátoch špecifických pre konkrétnu aplikáciu z pohľadu podpory vizualizačného jadra a jeho schopností pri práci s danou štruktúrou trojrozmerného modelu. Z pohľadu zobrazenia virtuálneho sveta je potrebný vhodný výber kamery a jej aplikovanie podľa druhu pohľadu, rovnako spôsobu interakcie. V skutočnosti existujú 2 základné typy reprezentujúce ortografické a perspektívne zobrazenie. V prípade odlišností týchto zobrazení je dôležité odlíšiť spôsob sledovania scény kamerou. Ortografické zobrazenie premieta obraz bez vnímania hĺbky obrazu, ktorý znemožňuje bližšie určiť objekty v popredí a v pozadí bez vykonania prekrývania. Perspektívne zobrazenie je najčastejšie používané v systémoch virtuálnej reality v dôsledku rozlíšiteľnosti hĺbky obrazu, ktoré je podstatné pri vytváraní stereoskopického obrazu. Podstatný vplyv vytvárajúci reálnejší dojem predstavuje výber vhodného druhu svetelných prvkov používaných v prostredí virtuálnej scény. Selekcia svetelných podmienok scény rozdeľuje dennú a nočnú scénu podľa prostredia a rozsahu definovanej scény. Denná scéna je najčastejšie reprezentovaná vysokým jasom scénických efektov, prípadnou kombináciou svetelných efektov. Nočná scéna obsahuje

početnejšie svetelné efekty s dôrazom pre osvetlenie objektov podľa aktuálnej priority v prostredí scény. Zohľadnením svetelných efektov a objektov scény je možné realizovať špeciálne efekty scény, závislé podľa typu prostredia a požiadaviek. SuperEngine umožňuje vytváranie video objektov alebo obrazových objektov viazaných na dvojrozmernú plochu v prostredí scény s možnosťou ovládania stavu prehrávania. Špeciálny efekt reprezentuje akýkoľvek dostupný alebo implementovaný efekt rozširujúci scénu v niektorom z jej stavov. Typické špeciálne efekty môžu obohacovať virtuálne prostredie o zvukové efekty obsiahnuté v trojrozmerných objektoch s viazaním rovnakej pozície alebo ich statickým umiestnením v scéne. Používanie efektov vplyva reálnejšie alebo prináša estetický efekt pre používateľa systému.

2.5.2. Voľba vlastností kamery

Aplikácia kamery v trojrozmerných prostrediach virtuálneho sveta okrem základného zobrazenia ortografickým alebo perspektívnym spôsobom obsahuje viazanie transformácií, rovnako zmeny parametrov vlastností obrazu. Kamera v prostredí SuperEngine disponuje viacerými parametrami so zohľadnením transformácií posunu, otáčania, definovania bodu orientácie smerového vektora, získavanie informácií o aktuálnom stave kamery na scéne, ovládanie orezávacej roviny s možnosťou definovania zorného uhla. V skutočnosti existuje viacero typov kamier s rozdielnym spôsobom implementácie pre jednotlivé scénické aplikácie. Kamera využívajúca pohľad na scény spôsobom oka kamery je pozorovateľom stavu scény bez ovplyvňovania jej parametrov pri akejkoľvek zmene vlastností scény. Druhým typom predstavujúcim pohľad zo strany dynamickej aplikácie prvku kamery zastupuje sledovanie z pohľadu orientovaného priamo očami používateľa, prípadne z pohľadu sledujúceho objekt zozadu. Pre každý typ kamery platí konkrétne využitie, možnosti sledovania aktuálneho diania z viacerých pohľadov alebo prepínaním kamery medzi jednotlivými časťami scény a typmi zobrazovania [31]. V prípade SuperEngine je problémová implementácia sledovania scény s využívaním skladaného obrazu viacerých kamier súčasne. Dôvodom je neschopnosť renderovacieho modulu Video Renderer rozdeľovať viaceré obrazy na jednom výstupe obrazovky v jednotnom čase. Z tohto dôvodu je možné sa medzi kamerami implementovanými na scéne len prepínať bez rozdelenia obrazu na viaceré súčasné fragmenty, napríklad spätné zrkadlá vozidla z pohľadu kabíny. V systéme virtuálnej jaskyne existuje niekoľko inštancií kamier, s rozdelením uhlov obrazu, viazaných na počet spustených inštancií zobrazovacieho modulu Video Renderer v rámci jednej výpočtovej jednotky. Je nutné zdôrazniť ich aplikáciu prostredníctvom využitia klastra počítačov, v ktorom každá výpočtová jednotka odbavuje niekoľko výstupov kamier pre obrazovky. V takomto prípade je možné dosahovať efekt viacerých kamier, nevýhodou je absencia zobrazenia viacerých obrazov kamier v jednom renderovacom okne inštancie Video

Renderer Win32. Systém SuperEngine využíva konfiguračné súbory pre nastavenie jednotlivých pohľadov kamier, ktoré sú obsiahnuté na každom renderovacom zariadení v klastrí zvlášť. Konfiguračné nastavenie nie je možné dynamicky meniť, reprezentuje statické nastavenie jednotlivých pohľadov a ovplyvňuje všetky scény súčasne. Akýmkoľvek zásahom do konfiguračného súboru niektorého z počítačov klastra môže byť poškodené riadenie poradia vykresľovania a delenia obrazu v systéme. Spätná konfigurácia je časovo náročná, rovnako zásah do konfigurácie systému z pohľadu zobrazovania neodporúča samotná spoločnosť Slovakia SuperComputers. Okrem nevýhod kamera disponuje širokou škálou transformácií so zohľadnením bodu orientácie (target) a smerového uhla. Smerový vektor kamery je možné posúvať priamo s bodom jej zobrazovacej roviny [Obr.10, A], pri ktorom je kamera posúvaná v smeroch do pravej a ľavej strany bez sledovania konkrétneho bodu na scéne. Druhý typ predstavuje kameru sledujúcu bod, pričom jej smerový vektor je orientovaný na miesto sledovaného bodu [Obr.10, B]. Rôzne typy transformácií kamery sú kombinovateľné a variabilne ovládateľné.

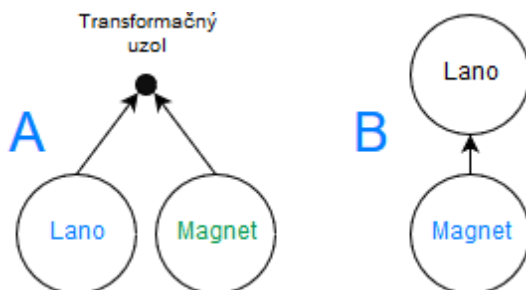


Obr.10 Posunutie kamery v smere osi y s bodom (A) a bez bodu orientácie pohľadu (B).

2.5.3. Transformácie objektov a uzlov

Dynamické vlastnosti je dôležité prenášať na akýkoľvek objekt obsiahnutý v scéne pre dosiahnutie interakcií a reálnejšieho vnemu z pohľadu používateľa. Každý dynamický objekt disponuje škálou transformácií v trojrozmernom priestore, ktoré je možné v SuperEngine kombinovať podľa požiadaviek a typu scény. Transformácie aplikovateľné na objekt scény obsiahnuté v systéme virtuálnej jaskyne umožňujú vykonávanie zmien pozície, rotácií, škálovania, sledovanie hraničných bodov v prípade nárazov, aplikáciu tieňovania, zvýraznenie objektu alebo viazanosti textúry. Rozdiely spočívajú vo viazaní objektov prostredníctvom transformačných uzlov, ktoré menia vlastnosti objektov z pohľadu

hierarchie. V prípade naviazania jedného objektu na druhý sa nadradený objekt vo väzbe stáva transformačným uzlom a objektom zároveň. Možné je aplikovať samostatný transformačný uzol, ktorý realizuje väzby medzi vybranou skupinou objektov v jednotnom čase [Obr.11, A]. Problém, ktorý nastáva z pohľadu aplikácie transformácie v prípade hierarchie transformačných uzlov je škálovanie alebo pohyby objektu na scéne z pohľadu stromu hierarchie. Príkladný opis uvažuje vytvorenie lana s použitím zaveseného magnetu, ktorý sa pohybuje spoločne s lanom [Obr.11, B]. Lano v prípade skupiny reprezentuje transformačný uzol, nadradený objektu magnetu. Aplikovaním transformácie na objekt lana sa súčasne aplikuje rovnaký druh transformácie na objekt magnetu. Takýto spôsob je korektný, pretože aplikovaním transformácie pre nadradený objekt, v tomto prípade transformačný uzol, sú ovplyvnené všetky nižšie zoradené objekty v hierarchii uzla. V prípade ak uvažíme aplikáciu transformácie pre objekt magnetu, neočakávame zmenu na transformačnom uzle, v tomto prípade lane. V prípade SuperEngine nastáva problém, ktorý v opačnom smere pri transformácii pozície magnetu alebo jeho veľkosti spätne transformuje rovnako objekt lana [Obr.11]. Takýto výsledok nie je korektný, nakoľko transformačný uzol môže byť ovplyvnený v prípade aplikácie transformácie len jeho nad-uzlom. Problém je riešiteľný vykonaním selekcie objektov a ich autonómneho správania, implementácia je v takomto prípade oveľa náročnejšia z pohľadu zložitosti scény.



Obr.11 Spôsoby viazania objektov pre aplikáciu transformácií.

2.5.4. Výber svetelných prvkov

Základná množina scénických svetelných prvkov v prostredí SuperEngine obsahuje bodové, reflektorové, plošné aplikácie svetelných podmienok. Bodové svetelné prvky reprezentujú svetelné lúče smerujúce z jedného definovaného bodu do všetkých ostatných smerov v trojrozmernom priestore. V prípade scénických aplikácií je použiteľné pre realizáciu svetelného efektu scény s využívaním rozptylu alebo priamym umiestnením v tele iného objektu, ktorý reprezentuje svetelný prvok (lampa). Reflektorové svetlo je charakterizované smerom a veľkosťou definujúcou priemer lúča s typickým tvarom svetelného kužeľa. Birn [31] považuje používanie reflektorových svetiel za vhodné

v prípade dynamických scén (automobilové svetlomety), ktoré využívajú definovanie smeru svetelného lúča s úpravou jeho vlastností. Rovnako je možné reflektorové svetlo používať v prostredí statickej scény pri zvýrazňovaní objektov v popredí. Plošný svetelný prvok obsahuje definovanú plochu, ktorej vychádzajúce lúče zvierajú so základňou plochy kolmý uhol. Plošné svetelné prvky je možné aplikovať v rôznych prípadoch stropných osvetlení, osvetlení scény z pohľadu rozsiahlejšieho svetelného zdroja alebo využívání efektov vyžarujúcich svetelných plôch (svietiace pásy, tuby). Častým problémom svetelných scén je nevhodná kombinácia svetelných prvkov s nepresnou parametrizáciou, v dôsledku ktorej je scénické svetlo veľmi výrazné, objekty nadobúdajú nereálne zobrazenie, prípadne je prostredie scénicky náročné vykresliť pre používanie generovania tieňov v rozpore s dopadajúcim svetlom iných svetelných prvkov na objekty trojrozmernej scény. V prípade SuperEngine je výpočtovo náročné využívanie viacerých kombinácií svetelných prvkov v reálnom čase. Aplikáciou vysokého počtu bodových svetiel s najnižším rozptylom za účelom eliminovať výpočtové nároky boli latencie viditeľné voľným okom. Problému je možné predchádzať riešením aplikácie textúr pre nočnú a dennú scénu selektívne s definovaním ich nasadenia priamo na plochy objektov pri základnom scénickom svetle generovanom priamo, pričom výsledný efekt bude čiastočne aplikovať vykresľovanie svetelných lúčov s kombinovaním svetelných plôch prostredníctvom tmavých a svetlejších častí obsiahnutých v textúre.

2.5.5. Okolie prostredia

Princípy vizualizácie moderných scén s docielením reálnejšieho vnemu zo strany používateľa využívajú rôzne typy okolí scén vytvárajúce neobmedzený priestor. Príkladná realizácia okolia spočíva v implementácii skybox prostredia spôsobom využívania panoramatických fotografií alebo generovaním objektov obsiahnutých v okolí formou oblohy a oblakov alebo iných scénických prvkov charakteristických pre typ vytváranej scény. Z pohľadu Isidora [32] výhodou využívania okolia scény nie je len estetickým obohatením celého prostredia, v prípade vhodnej kombinácie vyzdvihuje scénu s omnoho nižšími nárokmi na výpočtový systém oproti budovaniu nadbytočných objektov scény so zložitou štruktúrou, ktorá je v každom časovom momente vykresľovania scény rovnako zohľadnená. Okolie scény, takzvaný skybox, môže byť tvorené jedným objektom v tvare kocky, kvádra alebo gule s mapovaným povrchom pre umiestnenie panoramatickej textúry. Najčastejšie používané okolia reprezentujú tvar kocky alebo kvádra, prípadne často využívaným tvarom gule, ktorý dokonalejšie obklopuje prostredie scény. Najčastejší problém aplikácie kvádra alebo kocky predstavuje tieňovanie s aplikáciou nevhodného zatemnenia strán objektu skyboxu. V prípade jednoduchého okolia je možné použiť ambientnú zložku svetla pre zabezpečenia rovnomerného dopadu na strany okolia.

3. Návrh aplikácie v prostredí SuperEngine

Tvorba trojrozmerného prostredia pre aplikáciu v systéme virtuálnej jaskyne SuperEngine je závislá od nárokov kladených na zložitosť scény a druhu štruktúry používaných objektov. Kapitola v úvode rozoberá hlavné aspekty návrhu prostredia zamerané na zloženie scény a jej funkcionality. Z aspektu funkcionality scény rozoberá jednotlivé komponenty a správanie medzi nimi, pričom definuje postupnosť bodov pre správne overenie vzťahom medzi nimi. Prostredníctvom materiálov a textúr odlišuje postupy detailného spracovania mapy trojrozmerných modelov a jednoduchej mapy pri riešení nárokov kladených na detailný povrch modelu a rovnako nárokov kladených na výpočtový systém. V rámci reálnejšieho vnemu a používania efektových prvkov scény rozdeľuje svetelné prvky na úrovniach použitia s argumentovaním výhod a nevýhod pri aplikácii svetelného efektu scény. Z pohľadu implementácie scény definuje základnú konštrukciu skriptu s rozdelením jednotlivých sekcií vykonávajúcich sa pri nasadení skriptu scény v prostredí virtuálnej jaskyne a opisuje aplikovanie modulárneho viazania skriptov pri realizácii komplexnej scény. Záverom hodnotí vplyvy nesprávnej aplikácie skriptu viazaného na trojrozmernú scénu z pohľadu zamedzenia požadovaných funkcionalít alebo základných funkcií poskytovaných systémom.

3.1. Hlavné aspekty návrhu prostredia

Aplikácie scén v trojrozmernom prostredí obsahujú určitú parametrizáciu a rovnako požiadavky kladené pre splnenie požadovanej funkcionality. Niektoré prvky scény sú samostatné, iné viazané na ostatné objekty obsiahnuté v scéne podľa druhu aplikácie a požadovaného správania. Množina prvkov scény musí byť štruktúralne a funkčne odlišená prostredníctvom grafu scény so zobrazením jednotlivých vzťahov v hierarchii trojrozmerných objektov virtuálneho sveta. Dôvodom aplikácie grafu scény je dekompozícia z pohľadu rozdelenia objektov na základnej úrovni zobrazujúcej len hlavné časti bez zobrazenia hĺbky štruktúry, alebo grafom zobrazujúcim podrobné časti štruktúr s rozdelením na podrobné prvky jednotlivých objektov. Každé realizovateľné prostredie je možné zobraziť prostredníctvom grafu scény a definovať vzťahy súvisiace medzi jednotlivými objektmi. Príkladné je využívanie dynamickej scény, ktorá obsahuje objekt zložený z viacerých objektov, častí využívajúcich rôzne druhy transformácií v trojrozmernom priestore. Z pohľadu realizácie je objekt chápaný celistvo, napríklad model vozidla. Pri detailnom riešení sa naskytuje otázka závislá od aplikácie dynamiky, otáčania kolies, otvárania dverí, alebo sťahovania okien, ktorá vyžaduje model dekomponovať na jednotlivé funkčné časti. Graf scény sa v takomto prípade začína rozširovať, objekty využívajú viacero objektov hierarchicky nižšej úrovne a scéna začína byť na úrovni zložitosti náročnejšia. Podobne je

dôležité dbať na druh štruktúry, zohľadniť typ prostredia, formáty aplikovaných objektov a náročnosť scény pre dosiahnutie požadovaného vnemu používateľa.

3.1.1. Postup overenia funkcionality systému virtuálnej jaskyne

Vytváranie simulačných prostredí závisí od druhu aplikovanej problematiky, spôsobu použitia, rozsahu riešeného problému a technológií potrebných pre realizovanie simulovanej scény. Hlavným faktorom návrhu prostredia v rámci diplomového projektu je zameranie v oblasti testovania dostupných hardvérových prvkov, schopností vizualizačného jadra a skriptovacieho jazyka Ruby v softvéri systému SuperEngine. Prvý bod predstavuje realizáciu rôznych spôsobov transformácií trojrozmerných objektov s kompatibilnými formátmi ich štruktúr, vzhľadom na použitie jednoduchých alebo komplexných scén. Druhým bodom je reprezentácia vzťahov medzi trojrozmernými objektmi v prostredí virtuálneho sveta s aplikáciou rôznych interakcií, nárazov, hraníc a uzlov vrátane vytvorenia vzťažných a transformačných bodov scény. Tretí bod tvorby scény zahŕňa overenie svetelných a vizuálnych efektov scény pre vytváranie reálnejšieho dojmu v prostredí virtuálnej jaskyne, s aplikáciou dostupných inovácií prostredníctvom využívania rôznych metodík pri tvorbe svetelných prvkov. Hlavnou úlohou navrhovaného trojrozmerného virtuálneho prostredia je využívanie funkcionalít dostupných v systéme SuperEngine pre aplikáciu v riešeníach rôznych problematík. Z pohľadu ovládania je dôležité zdôrazniť používanie hardvérových prvkov zabezpečujúcich ovládanie interakcií medzi používateľom a systémom virtuálnej jaskyne alebo objektmi nachádzajúcimi sa v prostredí virtuálneho sveta. Prioritné je aplikovať vstupy realizované používateľom vrátane základných vstupných zariadení (klávesnica, myš) s kombináciou doplnkových zariadení umožňujúcich vykonávanie interakcie scény vzhľadom na pohyb používateľa v prostredí virtuálnej jaskyne (OptiTrack).

3.1.2. Úlohy aplikácie v prostredí virtuálnej jaskyne

Virtuálne prostredie navrhované pre testovanie funkcionalít, výkonu a reakcií systému virtuálnej jaskyne predstavuje realizáciu charakteristických úloh potrebných pre overenie správnosti funkcionalít a správania v systéme. Základné úlohy reprezentujú využitie transformácií objektov v priestore, využívanie transformačných scénických uzlov, viazanie funkcionalít vrátane interakcií aplikovaných pre množinu objektov, využívanie sady nástrojov dostupných z pohľadu hardvérového a softvérového hľadiska. V prípade využitia inovácií je dôležité overiť rozšírenie efektov virtuálneho sveta s aplikáciou aktuálnych trendov zameraných na tvorbu trojrozmerných objektov, svetelných prvkov a dynamiky scény. Základná množina úloh realizovaných v prostredí

virtuálnej jaskyne pre overenie schopností a reakcií systému je obsiahnutelná v scénach s predurčenou funkcionalitou:

- overenia transformácií objektov,
- interakcie používateľa a systému,
- interakcie objektov v závislosti od predurčených vlastností,
- vytvárania väzieb medzi objektmi.

Overenie transformácií objektov pozostáva z aplikácie jednotlivých transformácií v trojrozmernom priestore v rámci metód posunu, rotácie, škálovania mierky modelu s prípadnou realizáciou skosenia niektorého z rozmerov. Nasadenie transformácií je dôsledné aplikovať jednotlivo alebo kombinovane v závislosti od funkcionality trojrozmernej scény. Interakcie používania systému predstavujú základné vzťahy medzi používateľom a systémom prostredníctvom využitia hardvérových a softvérových prvkov, ktoré umožňujú vytváranie podnetov na systém z pohľadu používateľa a rovnako reakcie systému formou odpovede pre zvolenú formu vstupu. Interakcia medzi používateľom a systémom môže byť viazaná pre akúkoľvek funkčnú vlastnosť systému, najčastejšie je použitie viazané k dynamickým vlastnostiam trojrozmernej scény. Z pohľadu interakcie je dôsledné zahrnúť kolízie ovplyvňujúce reakcie objektov, rovnako správanie objektov pri zmene stavu scény. Reakcie sú opäť prenášané do dynamickej podoby, s možnosťou vykonávania rôznych transformácií, alebo statických reakcií realizujúcich zmenu vlastností objektu z pohľadu textúry alebo materiálu. Vytváraním väzieb medzi jednotlivými objektmi je možné vytvoriť rôzne typy trojrozmerných modelov, zložených z parciálnych častí. Príkladné je reprezentovanie zloženého modelu, ktorého časti sú usporiadané vo väzbách s definovanou prioritou. Typický model realizujúci väzby predstavuje dva alebo viaceré objekty viazané v určitom bode s aplikovaním určitej funkcionality. Podľa väzieb vzniká strom transformácií, zobrazujúci uzly a hierarchiu objektov. V prípade aplikácie transformácie na niektorý z nadradených modelov je automaticky transformácia prenášaná na podradený model. Väzby je možné kombinovať rôznym spôsobom s využívaním zložitých alebo jednoduchých transformácií podľa definovanej funkcionality systému. Dôležité je overiť schopnosť scény realizovať rôzne typy väzieb v závislosti od interakcií použitých v rámci dynamiky scénických prvkov. Súhrnne je takýto typ scén opísateľný prostredníctvom dynamických robotických ramien, alebo systémov, ktoré využívajú parciálne časti rotujúce v rôznych smeroch.

3.1.3. Rozdelenie scény

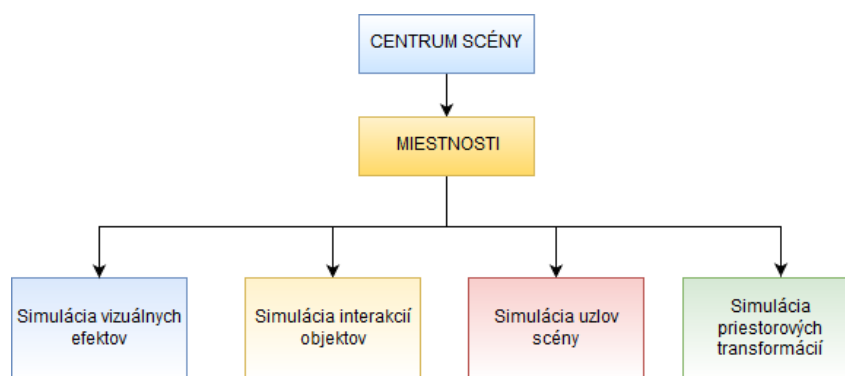
Aplikácia pre testovanie funkcionalít systému SuperEngine, určeného pre prácu s virtuálnym prostredím obsahuje modulárnu scénu zloženú z častí zameraných na realizáciu viacerých úloh s overením požadovaného typu funkcionality systému. V záujme využívania rozdielnych funkcionalít navrhovaná scéna reprezentuje základné prostredie viazané modulárne s ostatnými modulmi scén zameraných v oblasti testovanej problematiky trojrozmerného virtuálneho sveta. Dôsledné je overiť určené funkcionality a schopnosť skriptovacieho jazyka pri realizácii požadovanej funkcionality scény, realizácie reálnejšieho vnemu prostredia, zabezpečenia vlastností interakcií objektov, reakcií systému odpovedajúcich na podnety používateľa. Rovnako je dôsledné testovanie vizualizačného jadra spojeného s nárokmi kladenými na výpočtový systém z pohľadu vykresľovania scény v reálnom čase. Navrhnutá modulárna scéna realizuje uvedené požiadavky s porovnávaním jednotlivých vstupných parametrov s rozdelením na parciálne scény :

- centrum scény,
- simulácia vizuálnych efektov,
- simulácia interakcií objektov,
- simulácia uzlov scény,
- simulácia priestorových transformácií.

Centrum scény opisuje prvotný zobrazovací trojrozmerný priestor obsahujúci štyri vstupy pre uvedené simulačné moduly. Každý zo vstupov umožňuje používateľovi navštíviť zvolený virtuálny priestor so špecifickou funkcionalitou. Simulácia vizuálnych efektov zobrazuje vstup obsahujúci prostredie s miestnosťou v tvare prstenca pre demonštráciu rôznych kombinácií efektivej scény s využitím realistických textúr a materiálov modelov, svetelných prvkov v použití inovácií s možnosťami generovania textúr prostredníctvom využitia modelovacieho nástroja. V prípade realizácie scény pre simuláciu interakcií objektov je aplikované prostredie žeriavu s možnosťou prenášania nákladu s využitím hardvérových vstupov prostredníctvom zariadenia Joystick. Jednotlivé funkcionality sú rozdelené podľa spôsobu manipulácie s nákladom a posúvaním kabínou žeriavu. V rámci simulácie uzlov scény je predvolené prostredie stanice s obsiahnutým zariadením hydraulického ramena skladajúceho sa z viacerých častí transformovaných prostredníctvom využívania kinematiky bodov. Finálna scéna s využívaním priestorových transformácií opisuje kombináciu transformácie objektov v priestore s možnosťou vytvárania jednoduchého transformačného reťazca alebo zložitej transformácie viazaných objektov prostredia a kamery.

3.1.4. Graf scény

Základná modulárna scéna obsahuje štyri scény s realizáciou vybraných funkcionalít pre overenie a testovanie schopností softvéru SuperEngine. V prípade hlavného grafu scény [Obr.12] sú posudzované jednotlivé časti reprezentujúce ďalšie pod úrovne scén aplikovaných v rámci testovania. V predošlých podkapitolách boli scény opísané z pohľadu realizácie, prípadne riešenia úloh, pre ktoré boli navrhnuté. Dôležité je rovnako poznamenať zloženie scény z hľadiska obsiahnutých objektov trojrozmerného prostredia. Hlavnú časť každej scény predstavuje základné scénické prostredie s kompozíciou jednotlivých objektov v určenom poradí podľa definície cieľového modelu.



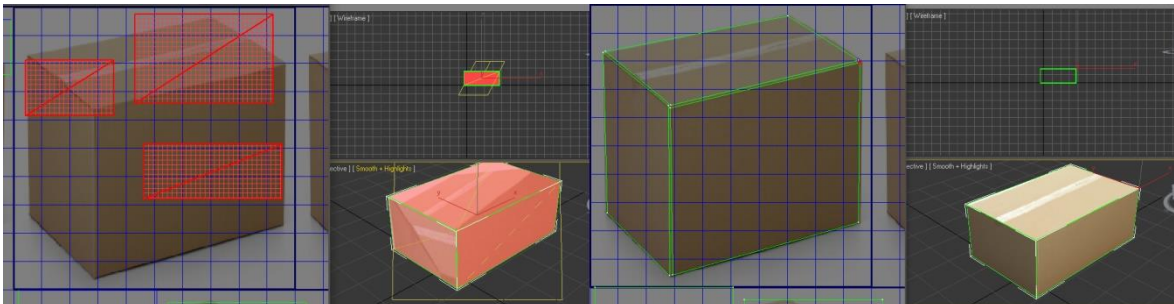
Obr. 12 Graf scény s obsahom modulov.

Každá z modulárnych scén obsahuje detailnejšie prostredie pre testovanie konkrétnej problematiky funkcionalít systému virtuálnej jaskyne. V prípade realizácie modulárnych scén sú grafy spracovávané detailnejšie, obsahujú jednotlivé prvky objektov, viazanie transformačných uzlov v hierarchii podľa potrieb simulácie scény. Detailnejší graf scény opisuje členitosť objektov pozostávajúcich z viacerých modulárnych častí s viazanou štruktúrou alebo objekty zložené prostredníctvom delenia na podradené a nadradené čiastkové modely.

3.1.5. Materiály objektov

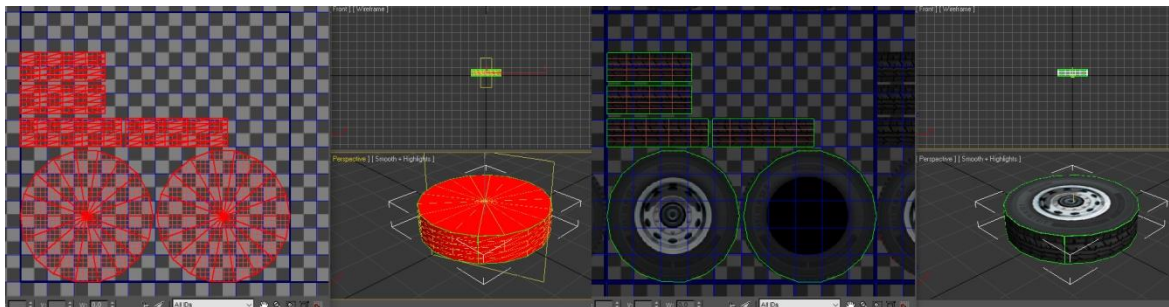
Scény realizované v prostredí SuperEngine obsahujú rôzne typy používaných textúr vrátane textúry s obsiahnutými svetelnými prvkami. Hlavný postup pri vytváraní textúry viazanej na povrch telesa je správne vytvorenie (mapping) siete telesa pomocou modelovacieho nástroja. V prípade riešenia diplomového projektu bol aplikovaný modelovací nástroj 3ds Max 11 [33] s dostupnou licenciou na FEI TUKE. Sieť telesa opisuje jeho štruktúru plôch aplikovaných v dvojrozmernom pohľade pripomínajúcu vystrihovačku. Každá plocha má definované vlastné rozmiestnenie a tvar podľa jej veľkosti v trojrozmernom modeli. Dvojrozmerné zobrazenie siete alebo mapu povrchu modelu je možné považovať za papier, na ktorom sú časti vystrihovačky – časti modelu obsiahnuté. Veľkosť siete povrchu trojrozmerných objektov v scéne závisí od tvaru

a zložitosti jeho jednotlivých plôch, prípadne podľa individuálnej potreby pri prioritnom určení detailnejších a jednoduchších plôch telesa. Povrch plochy je možné v mape siete zmenšiť alebo zväčšiť, skosiť alebo upraviť podľa nanášanaj textúry. Každá textúra je aplikovateľná rôznymi spôsobmi závislými podľa formy nanášania na trojrozmerný objekt scény. V praxi je možné využitie dvoch základných spôsobov nanášania textúr s využívaním aplikácie prostredníctvom výrezu z textúrového obrazu alebo vytvorením textúry povrchu objektu podľa vygenerovanej mapy. Výrez textúrového obrazu reprezentuje akýkoľvek obraz nanesený na povrch telesa formou textúry s využívaním škálovania, rozmiestnenia a rotácie siete povrchu telesa. Pri postupe nanášania textúry prostredníctvom výrezu je použitý textúrový obraz zachovaný v pôvodnom tvare, ktorému sa prispôsobuje sieť telesa, charakteristický fotografiou alebo rôznym grafickým obrazom [Obr.13]. Príkladné je používanie fotografie, ktorá zachytáva reálny objekt so zámerom preniesť jeho textúru na trojrozmerný model v grafickom nástroji. V takomto prípade je mapa siete trojrozmerného objektu prispôbená tvarovo a rozmerovo fotografii alebo grafickej reprezentácii použitej formou textúry. Náročnosť pri prispôbení siete povrchu trojrozmerného modelu textúre obsahujúcej fotografiu je časovo najmenej náročná, nevyžaduje si komplikované riešenie alebo zásah prostredníctvom grafického nástroja pre úpravu fotografie, prípadne grafiky.



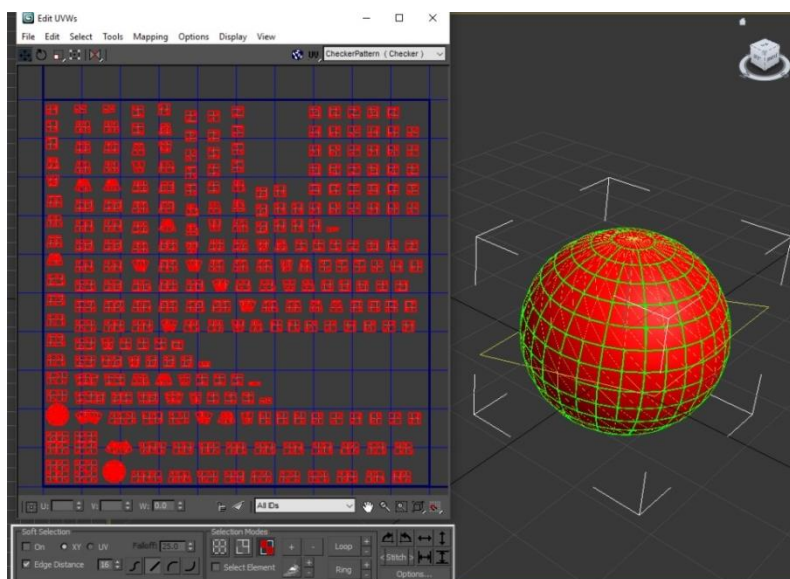
Obr. 13 Prispôsobenie siete modelu v modelovacom nástroji.

Aplikácia vytvorením textúry povrchu objektu podľa vygenerovanej mapy objektu predstavuje vytvorenie textúrového obrazu, ktorý je prispôbený sieti trojrozmerného modelu pomocou využitia grafického nástroja. Prvým krokom je vytvorená sieť modelu, bez úprav jej tvaru alebo škálovania rozmerov [Obr.14]. Príkladné je realizovanie objektu, pre ktorý je náročné aplikovanie fotografie alebo obrázka z dôvodu náročných povrchov, prípadne tvarov objektu. V takomto prípade je jednoduchšie pripravovanie textúry podľa siete povrchu objektu s využitím grafického nástroja a vygenerovaním jeho mapy.



Obr. 14 Prispôsobenie textúry v grafickom nástroji.

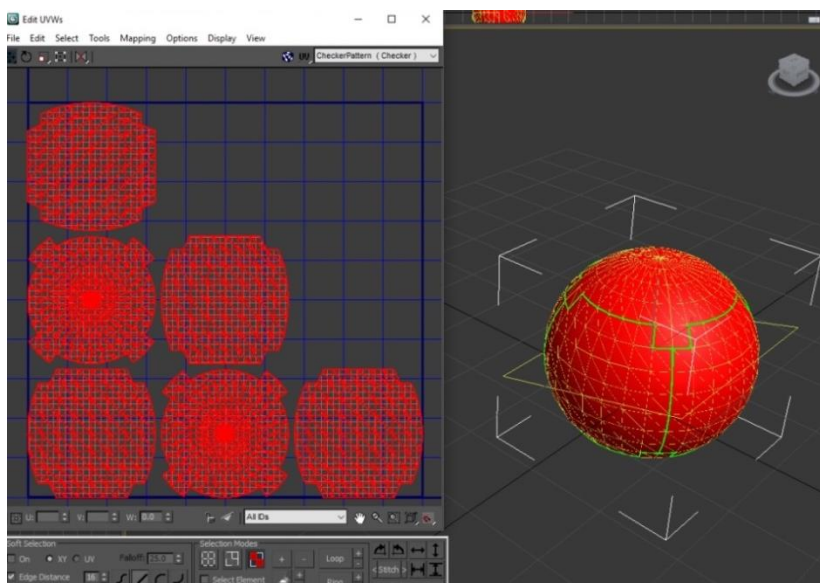
V prípade realizácie zložitejších štruktúr je často používaná metóda pre vyhladenie častí povrchov a navrhnutí siete modelu z pohľadu komplexnejšieho rozdelenia (flaten mapping). Takéto využitie v opačnom prípade prináša reálnejšie zobrazenie modelu bez výrazných chýb v sieti a nanášania textúry [Obr.15]. V procese vytvárania textúry je každá plocha telesa vyčlenená zvlášť, bez prekrývania siete povrchu modelu. Vyhladenie siete objektu je z pohľadu nárokov na výpočtový systém náročnejšie v prípade využívania vysokého rozlíšenia textúr, pri ktorom je potrebné sledovať každú z plôch objektu a dosahovať zobrazenie textúry korektne.



Obr. 15 Aplikácia siete s detailným rozdelením plôch objektu.

V prípade dosahovania nižších nárokov na výpočtový systém je možné realizovať vytvorenie siete prostredníctvom krabicového rozdelenia (box mapping), ktoré navrhuje sieť objektu zo všetkých základných pohľadov aplikovaných zo spodnej časti, vrchnej časti, prednej a zadnej, rovnako pravej a ľavej časti objektu kolmým pohľadom. Takýto typ siete predstavuje jednoduché rozdelenie povrchov modelu pre aplikovanie textúry plôch. Nevýhodou vytvárania siete modelu pomocou aplikovania metódy krabicového rozdelenia povrchu trojrozmerného telesa je nesprávne definovanie rozmerov alebo zakrivenia plôch. Nepresnosť vzniká pri deformácii

pohľadu mapovania na objekt, pričom trojrozmerný objekt nepredstavuje len jednoduché primitíva formou kocky alebo ihlanu. V prípade generovania siete gule je tvar plôch z rôznych pohľadov kamery rozdielny. Plochy nachádzajúce sa bližšie k pohľadu kamery mapovania sú detailnejšie, obsahovo väčšie, bez výrazných deformácií [Obr.16]. Opakom sú plochy, ktoré obkolesujú povrch gule v pohľade mapovania zo strán, sú výrazne skosené a deformované. Takýto typ mapovania môže výsledne spôsobiť nepresnosti medzi rozsahom a veľkosťou nanášanaj textúry, z opačného aspektu je výhodný pre kladenie nižších nárokov na výpočtový systém pri vykresľovaní trojrozmerného objektu na scéne.



Obr. 16 Aplikácia siete s jednoduchým rozdelením plôch objektu.

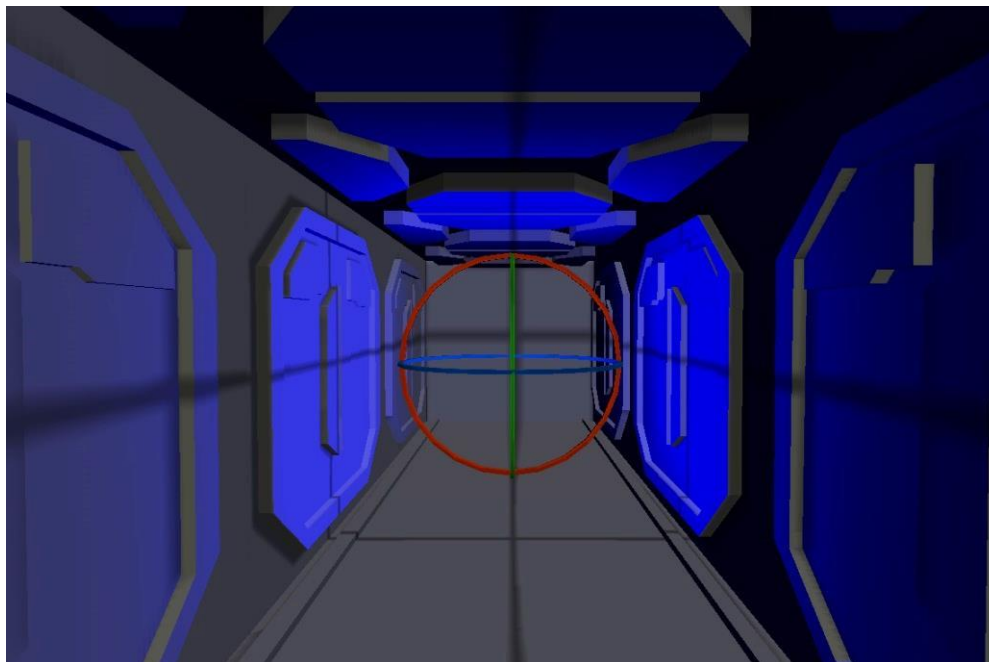
Nevýhodou SuperEngine je aplikovanie dvoch rozdielnych materiálov s textúrou na jeden objekt scény. Zobrazenie textúry bolo na objekte v spomínanom prípade chaotické, neusporiadané alebo nebolo aplikované, pričom bol objekt viditeľný v scéne s povrchom sivej farby bez obsahu textúry. V niektorých prípadoch bolo potrebné objekt dekomponovať na jednotlivé plochy správajúce sa autonómne, plochy boli samostatné objekty skladajúce model. Výsledok preukázal schopnosť správne zobrazovať textúry prostredníctvom modelov s navrhnutou sieťou, v ktorej boli konkrétne materiály podľa potrieb obsiahnuté v jednej textúre. Samotný SuperEngine umožňuje zmenu textúr viazaných na objekt počas vykonávania scény. V takomto prípade boli prejavované chyby zobrazenia, nakoľko SuperEngine nedokázal textúry zamieňať, len ich prekrývať. Tieto chyby sa môžu podieľať na problémoch v prípade, ak model v niektorom z momentov alebo udalostí vyskytujúcich sa na scéne zmení textúrový podklad. Predstavme si vytvorenie objektu reflektorového svetla v modelovacom nástroji, ktorému pridáme textúru vo forme červenej farby a vypnutom stave, pričom bude červená farba obsahovať tmavý odtieň. V prípade

skriptovania rátame s možnosťou zmeny textúry, ak je reflektor činný, textúra bude reprezentovať typ zvýraznenej červenej farby. Problém nastáva pri práci s modelom v prostredí SuperEngine, nakoľko je nereálne dosiahnuť efekt zmeny textúry bez viditeľného prelínania dvoch textúr súčasne. Takýto problém je riešiteľný ešte jednou metódou, spomaľuje ale chod scény, nakoľko zahŕňa objekt navyše. Riešiteľnosť predstavuje využívanie viditeľnosti objektov, ktorých štruktúra je na začiatku scény načítaná, ale viditeľný je len aktívny objekt. V prípade nečinnosti objektu reflektorového svetla je aplikovaná textúra v nezapnutom stave, v prípade aktivovania svietivosti je viditeľnosť objektu reflektora v nezapnutom stave nulová a mení sa viditeľnosť objektu reflektora v zapnutom stave na stav viditeľnosti. V prípade používania jednoduchých modelov s nízkym obsahom polygónov v štruktúre je takýto typ scény realizovateľný, v opačnom prípade so zložitými štruktúrami sú výpočtové nároky na systém oveľa vyššie.

3.1.6. Svetelné prvky scény

SuperEngine disponuje troma základnými typmi svetiel s rozdelením na bodové, smerové a reflektorové svetlo. Každý z typov vytvára rozdielny druh efektu na scéne, závislý podľa potrieb umiestnenia svetla v priestore, parametrami pre smerový vektor, intenzitu, farby vrátane difúznej a ambientnej zložky svetla. Výhodou používania scénických prvkov svetelnej scény je aplikácia v prostredí, ktoré požaduje zmenu svetelnej scény alebo dynamické transformácie svetelných prvkov vrátane posunu a zmeny rotácie. Takéto typy práce so svetelnými prvkami sú bežné v každom vizualizačnom jadre obsahujúcom svetelné prvky, efekty. Dôsledné je poznamenať porovnanie používania svetelnej scény s pohľadom na nároky výpočtového systému. Svetelné prvky predstavujú rôzne druhy svetiel používaných v trojrozmernom priestore s určitou funkcionalitou. Jednoduchý typ smerového svetla umožňuje definovať bod svetelného zdroja a vektora, v prípade reflektorového svetla je možné definovať rozmiestnenie plochy a dopadu lúčov s možnosťou nastavenia útlmu svetla vzhľadom na vzdialenosť dopadu. Bodové svetlo predstavuje typ svetelného efektu s používaním bodu so šírením svetelných lúčov v každom smere od bodu vyžarovania. Typ jednoduchého smerového svetla je často aplikovaný v prípadoch scény, ktoré využívajú smer prúdenia svetla na definované miesto, objekt. Smerové svetlo je často aplikované v podobe slnka pre dosahovanie výrazných tieňov na scéne, pričom je priemer lúča smerového svetla niekoľko násobne väčší oproti obsahu podstavy scény. Dôvodom je tvar šíreného lúča z pohľadu profilu, ktorý predstavuje kužeľ rozširujúci sa smerom od zdroja svetla. Reflektorové svetlo je výrazne blízke typu smerového svetla, nakoľko aplikuje podobný spôsob šírenia svetelných lúčov v priestore. Tvar prúdenia lúča je reprezentovaný valcom s rovnakým priemerom pri zdroji a dopade na trojrozmerný objekt scény. Typickým príkladom bodového svetelného prvku

je žiarovka bez objímky, len svetelná guľa šíriaca svetlo do okolia. Lúče sa od bodového svetla šíria v každom smere kolmo na bod zdroja reprezentovaného tvarom blízkeho gule [Obr.17]. Nevýhodou svetelných prvkov virtuálneho prostredia SuperEngine je neschopnosť zobrazovania zdroja svetelného prvku, v prípade ktorého je náročné realizovať scény s využívaním štruktúry formou zobrazenia lúčov pre reálnejší vnem scény.



Obr. 17 Neprirodzené šírenie svetla v priestore.

V takomto prípade boli nasadené scény v prostredí virtuálnej jaskyne nereálne a umelé, spôsobovali spomalenie výpočtov systému pri vykresľovaní scény, znemožňovali vykonávanie interakcií bez výrazných latencií. Možnosť riešenia problému spočívala v statickom vykresľovaní svetla na povrchy textúr prostredníctvom využitia svetelných prvkov modelovacieho nástroja s možnosťou vytvorenia svetelnej mapy povrchu telesa. Takéto riešenie reprezentuje najnižšie nároky na výpočtový systém z pohľadu náročnosti aplikovaných štruktúr, dosahuje reálnejší vnem a umožňuje používanie vysokého počtu svetelných prvkov v reálnom čase. Nevýhodou je zobrazovanie svetelných efektov staticky bez generovania tieňov, rovnako bez možnosti aplikácie transformácie posunu v rámci scény [Obr.18]. Nie je možné pochybovať o chýbajúcej zložke vlastností svetelných efektov v prostredí SuperEngine z pohľadu zobrazenia zdroja svetelného prvku a smerového lúča vo viditeľnej podobe. V opačnom prípade vykresľovane svetelných máp predstavuje výhodné riešenie dosahovania reálnejších efektov.



Obr. 18 Porovnanie scény s generovaním svetla vizualizačného jadra a svetelnou textúrou.

3.1.7. Interakcie objektov scény

Interaktívne prostredia virtuálnych jaskýň obsahujú rôzne aplikácie interakcií medzi objektmi scény s rôznou funkcionalitou pre prácu vo virtuálnom prostredí. V dôsledku testovania bolo potrebné overiť základné interakcie objektov na scéne s využívaním možností ohraničenia prostredníctvom nárazových plôch a algoritmického riešenia implementácie metód pre zabezpečenie overovania kolízií. SuperEngine ponúka metódy pre získavanie súradníc kvádra obalujúceho objekt v jeho súradnicových maximách z každej strany (bounding box). Takéto maximá dovoľujú definovať krajné body objektu v trojrozmernom priestore s využívaním usporiadaných dvojíc súradníc objektu obaleného kvádom. Výstup zaručuje osem základných dvojíc z pohľadu podstavy a vrchnej steny obalovaného objektu opisujúce vrcholy. Každý z vrcholov opisuje tri základné vlastnosti rozmerov z pohľadu výšky, šírky a hĺbky v priestore scény. Interakcie súvisia s reakciami objektov pri výskyte niektorej z udalostí v simulovanom prostredí s prejavom odpovede podľa požadovanej formy. Medzi interakcie scény boli zaradené okrem kolízií prenášanie vlastností objektov pri náraze, deformácia stien určená statickým modelom. Náraz spôsobuje prenášanie pohybových vlastností objektu vykonávajúceho akciu na objekt vykonávajúceho reakciu v prostredí, prípadne deformáciu oboch objektov pri náraze. Testovaním v scéne simulácie interaktívneho prostredia opisujú modelové situácie schopnosti implementácie docieľiť niektorú z požadovaných vlastností. Interaktivita scény môže byť rovnako viazaná prostredníctvom vstupu tlačidla v používateľskom rozhraní JavaConsole s viazaním na zmenu stavu scény podľa potrieb.

3.2. Realizácia skriptovania scény

Vytváranie scén pre aplikáciu v prostredí systému virtuálnej jaskyne je potrebné realizovať pomocou skriptov implementovaných v jazyku Ruby 1.8.6. Každý zo skriptov scény obsahuje definované sekcie, zamerané na oddelenie častí charakterizované sekciami vykonávania. Programátor môže prostredníctvom skriptu vytvárať rôzne interakcie v systéme, riadiť a transformovať trojrozmerné modely alebo pracovať s dostupnými prvkami systému vrátane vytvárania používateľských rozhraní. Každý zo skriptov dodržiava definovanú konštrukciu podľa ktorej je riadenie rozdelené na jednotlivé úrovne v prípade vykonávania slučiek, inicializácie zariadení, pozastavenia alebo ukončenia scény. Využívanie skriptovacieho jazyka Ruby je nenáročné pri realizovaní trojrozmernej scény, používaním dostupnej množiny metód od spoločnosti Slovakia SuperComputers. V rámci používania metód je možné vytváranie objektov, aplikovanie transformácií, pracovanie so štruktúrami trojrozmerných objektov. Implementovanie skriptov umožňuje modulárne viazanie tried vytvárajúcich objekty virtuálneho sveta alebo rôznych abstrakcií podieľajúcich sa riadení scény. Scénu je nutné vytvárať s dohľadnutím na možné chyby pri implementovaní skriptu, nakoľko softvér SuperEngine absentuje nástrojom pre skriptovanie s kontrolou syntaxe jazyka a metód.

3.2.1. Základná konštrukcia skriptu

Vytváranie fragmentov skriptov scén je realizované využívaním jazyka Ruby so základnou konštrukciou a aplikáciou metód vytvorených spoločnosťou Slovakia SuperComputers. Pre správne načítanie balíčka scény je potrebné vytvorenie súboru „názov.rb“ s implementovanou konštrukciou skriptu v definovanom tvare. Každý skript implementovaný pre systém virtuálnej jaskyne aplikuje štyri základné sekcie pre načítanie scény a scénických objektov, vykonávania scénickej slučky, aktivít aplikovaných pri pozastavení scény a aktivít vykonávaných pri ukončení scény. Skript môže jednotlivé sekcie obsahovať v akomkoľvek poradí, bez určenia postupnosti vykonávania, dôležité je zachovať tvar a správnosť vyčlenenia miesta kódu, pre predchádzanie konfliktom spôsobujúcich nechcené reakcie prostredia alebo blokovanie a nevyžiadané zastavenie scény. Nevýhodou nekontrolovaného riadenia scény prostredníctvom nesprávne zapísanej konštrukcie alebo volania niektorých z metód v nedefinovanom tvare spôsobuje časté problémy pri spustení scény. Najčastejšie komplikácie objavujúce sa v dôsledku nesprávnej aplikácie konštrukcie alebo vyššie spomínaných chýb, spôsobujú nečakané reakcie systému virtuálnej jaskyne, charakteristické nevhodnými odozvami systému na podnety vstupujúce zo strany používateľa. Využívaním zariadení pre vstupy z klávesnice, myšky alebo sledovaním pohybu bodov prostredníctvom systému OptiTrack je nutné dbať na správne volanie zariadení v skripte. Pre dosiahnutie správneho chodu skriptu je nutné implementovať každú

sekcii, v prípade jej nepotrebnosti je ponechané prázdne telo sekcie. Sekcia musí byť v každom prípade obsiahnutá v skripte, v opačnom prípade je možné očakávať nechcené reakcie zo strany systému.

3.2.2. Sekcie skriptu riadenia scény

Skript trojrozmernej scény pozostáva zo štyroch základných sekcií realizujúcich načítanie scény, aktivitu vykonávanú v slučke, pozastavenie a ukončenie scény v prostredí virtuálnej jaskyne. Každá sekcia je charakteristická definovaním jej ohraničenia pre vykonávanie prostredníctvom zápisu kódu v charakteristickom tvare. Sekcia charakteristická pre definovanie premenných, riadenia prístupu, zápisu metód alebo načítaním štruktúry trojrozmerných objektov využíva zápis mimo slučky, ktorý nemusí dodržiavať presné postupy. V prípade načítania skriptu je každý riadok kódu obsiahnutý mimo sekcií slučiek automaticky považovaný za fragment s obsahom údajov, ktoré je potrebné načítať pred spustením scény. V dôsledku rozdelenia sekcií je skript oveľa členitejší ale zároveň prehľadnejší. Z pohľadu akcií vykonávaných v skripte je možné používanie rôznych sekcií pre rôzne akcie. Sekcie scén je nutné zaradiť v každom zo skriptov, v opačnom prípade nie sú scény spustiteľné, nakoľko sú sekcie priamo viazané pre spúšťanie prostredníctvom JavaConsole. Prvotná sekcia obsahuje funkcionality spustenia scény, ktorá spúšťa vykonávanie hlavnej slučky akcií, zabezpečujúcu dynamiku a cyklické vykonávanie. Sekcia pre štart scény využíva nasledujúci zápis v skripte:

```
def ENGINE::OnSceneStart ()  
  
# telo vykonávanej slučky  
  
end
```

Spustenie scény dovoľuje používateľovi aktivovať dynamiku alebo cyklické vykonávanie obsiahnuté v skripte, prípadne spustiť sledovanie vstupných zariadení. Častým problémom pri používaní sekcie pre spustenie scény „OnSceneStart“ je zápis kódu v jej tele, ktorý využíva cyklické opakovanie, pričom zabraňuje prechodu do sekcie „Actions“. Hlavný fragment reprezentujúci slučku vykonávania akcií je definovaný ohraničením :

```
def ENGINE::Actions  
  
# telo vykonávanej slučky  
  
end
```

V prípade tela vykonávanej slučky „Actions“ je dôležité zahrnúť všetky prípady, pre ktoré je potrebné sledovať stav, reagovať na udalosť, alebo plynule aplikovať niektorú z metód, prípadne transformácií. Telo fragmentu pre akcie zodpovedá za cyklické vykonávanie

obsiahnutého kódu, rovnako prispieva k monitorovaniu stavu rôznych premenných. Riadenie scény v slučke „Actions“ reprezentuje hlavnú funkčnú zložku skriptu, ktorá dokáže operovať nad trojrozmerným prostredím. Najčastejším problémom pri vykonávaní je nekorektné používanie metód alebo vstupov s častým zablokovaním scény, pre ktoré je potrebné scénu opakovane reštartovať. Podobne je možné scénu pozastaviť prostredníctvom využívania sekcie :

```
def ENGINE::OnScenePause()  
  
# telo vykonávanej slučky  
  
End
```

Sekcia pre pozastavenie scény dovoľuje používateľovi scénu pozastaviť a vykonávať telo s obsahom zapísaného kódu. Pozastavením scény je možné používať interaktívne správanie objektov prostredníctvom cyklického vykonávania tela sekcie. Nevýhodou pri pozastavení scény je situácia pri využívaní vstupných zariadení, ktoré naďalej zasielajú informácie do systému. Problém sa opakuje len v prípade ak sú zariadenia inicializované priamo v slučke hlavného vykonávania akcií „Actions“ miesto inicializácii mimo slučky. Ďalší z fragmentov sekcií reprezentuje ukončenie scény, ktorej vykonávanie prebieha v slučke prostredníctvom zápisu:

```
def ENGINE::OnExit  
  
# telo vykonávanej slučky  
  
end
```

Telo sekcie pre ukončenie scény môže rovnako ostať prázdne, v takomto prípade bude scéna ukončená priamo bez plynulého prechodu. Rozdiel medzi plynulým a priamym ukončením scény nespôsobuje výrazné rozdiely. Dôležité je dodržať spomínaný fragment sekcie pre správne spustenie skriptu scény.

3.2.3. Modulárne viazanie skriptov

Využívanie modularity pri viazaní prvkov alebo častí scén z pohľadu používania skriptovacích modulov je výhodné v prípade, ak zložitost scény je náročná, prípadne obsah kódu je členitejší a náročnejší. Každý reprezentujúci model alebo funkcionality zvyšuje náročnosť zápisu skriptu, prípadne zvyšuje členitosť jeho kódu. V dôsledku dosiahnutia lepšej prehľadnosti je modulárne spájanie výhodné, nakoľko umožňuje vytváranie skriptov s použitím tried objektov, alebo operácií a funkcionalít, ktoré sú následne volané v hlavnom skripte scény „scene.rb“. Rozdielom medzi hlavným súborom a súborom triedy skriptu je zápis sekcií riadenia skriptu pre vykonávanie, pozastavenie alebo ukončenie scény. V prípade hlavného súboru „scene.rb“ je nevyhnutné využívanie sekcií riadenia, nakoľko reprezentuje jadro celého riadenia scény.

Súbory tried predstavujú moduly volané hlavným súborom scény, preto nie je možné dopĺňať sekcie riadenia scény priamo v tele kódu triedy. Trieda obsahujúca akúkoľvek sekciu riadenia spôsobuje pri volaní jej objektu okamžité zastavenie a prerušenie vykonávanej scény. Súbory tried môžu obsahovať len metódy a premenné pre konštrukciu jej objektov. Najčastejším problémom je viazanie funkčne odlišných scén v jednom balíčku. Je logické, že používanie skriptu s náročnou štruktúrou je výhodné rozdeliť na modulárne časti a volať len vybrané objekty tried v potrebných prípadoch použitia na scéne. Pomocou aplikácie viacerých scén v rámci jedného balíčka nastáva problém z dvoch pohľadov realizácie. Myšlienka uľahčenia pre používanie odlišných scén v jednom balíčku so vzájomným previazaním by mohla byť uskutočnená v prípade, ak by boli scény obsiahnuté každá vo vlastnej triede a volané len ich inštancie. Problémom je nemožnosť dosadenia sekcií riadenia do triedy, nakoľko trieda môže obsahovať len premenné a metódy, v tom prípade nie je možné triedu spustiť ako scénu. Scéna je reprezentovaná hlavným súborom, trieda je rovnako reprezentovaná skriptom, ale nemôže byť považovaná za hlavný súbor. Dôvodom sú sekcie riadenia, ktoré spôsobujú problém pri riadení dvoch scén súčasne. Predstavme si dve rozličné scény s rozdielnou funkcionalitou, obsahujúce sekcie riadenia pre spustenie, vykonávanie, pozastavenie a ukončenie. Každá týchto scén využíva vlastnú logiku riadenia, z pohľadu správania trojrozmerných objektov, definovaných premenných, využívania vstupov. Keďže je používanie logík odlišné, je rovnako odlišné používanie objektov na scéne alebo vstupov, čím sa mení riadenie scény. Ak by sme chceli dosiahnuť efekt a vytvoriť dve scény priamo v jednom hlavnom súbore, prudko by narástla členitosť a zložitosť kódu, výskyt chýb pri rovnakých názvoch premenných, prípadne ich prepísaniu. Rovnako by sa prejavili latencie systému pri vykresľovaní scény a nárast nákladov na výkon systému. Scény je možné spájať s výskytom vyššie spomínaných rizík a nevýhod, alebo s možnosťou optimalizácie členitosti kódu len v prípade, ak sú scény jednoduché a používajú nízku úroveň funkcionalít. Najvhodnejším spôsobom je aplikovanie modulov scény s rovnakou logikou a používaním vstupov so špecifickou funkcionalitou, bez zmeny počas riadenia scény. V opačnom prípade je spájanie dvoch scén nevýhodné a často nemožné.

3.2.4. Vplyv nesprávnej aplikácie skriptu

Častým problémom nesprávnej funkcionality scény je chyba vyskytujúca sa v skripte, spôsobujúca niektorú z komplikácií. Komplikácie predstavujú rôzne druhy obmedzenia funkcionality systému z pohľadu jeho správania. Chyby vyskytujúce sa v nesprávnom volaní vstupných zariadení spôsobujú najčastejšie zastavenie scény s odopretím rôznych dynamických vlastností, najčastejšie zamedzeniu transformácií trojrozmerných objektov. Scéna je v takomto prípade viditeľná, nie je ale interaktívna a rovnako nereaguje na podnety zo strany používateľa.

Balíček scény môže byť podobne nespustiteľný v prípade ak obsahuje niektoré nesprávne zapísané premenné alebo aplikuje volania premenných, ktoré v skripte nie sú obsiahnuté. Pokiaľ nie je scéna spustiteľná, je značným dôvodom chyba vyskytujúca sa v skripte scény, prípadne absencia niektorého z modulov tried v balíčku scény. Ďalšou z možností výskytu chybovosti systému vzniká volaním zariadenia, ktoré nie je pripojené alebo je volané nesprávne. V uvedenom prípade nie je systém virtuálnej jaskyne schopný nadviazať spojenie pre očakávaný vstup, nakoľko zariadenie neexistuje. Nastáva zablokovanie scény prostredia bez akýchkoľvek reakcií na podnety používateľa, scéna musí byť ukončená. Ukončenie scén nie je vždy korektným riešením pri výskyte chýb v systéme. Chyby spôsobené nesprávnym volaním vstupných zariadení sa často podieľajú na zablokovaní riadenia vizualizačného jadra, pričom sa chybovosť preukazuje aj na ostatných scénach. Takýto problém je rozsiahly, vizualizačné jadro neustále očakáva nekorektné vstupy. V dôsledku opätovného sprevádzkovania správnej funkcionality je potrebné vykonať reštartovanie vizualizačného jadra. Vytváranie skriptov pre prostredie virtuálnej jaskyne spôsobuje v kontrole správnosti problémy, nakoľko softvér SuperEngine neobsahuje skriptovací nástroj s možnosťou kontroly syntaxe zápisu kódu alebo kontrolou správneho volania metód, premenných alebo zariadení. Voľba výberu skriptovacieho nástroja ostáva na používateľovi, komplikáciou pri kontrole správnosti zápisu kódu je množina používaných metód pre riadenie scény z pohľadu SuperEngine, ktorú musí používateľ kontrolovať svojpomocne bez automatizácie kontroly zápisu. V prípade náročných scén je komplikované hľadanie chýb a najvhodnejším spôsobom pre používateľa je komponovanie scény pomocou malých krokov a pravidelným kontrolovaním implementácie v skripte. Časová náročnosť v uvedenom prípade pri realizácii akejkoľvek scény prudko narastá. Vplyv nesprávnej aplikácie skriptu sa prejavuje v rôznych podobách, dôležité je dbať na správnosť zápisu implementácie.

4. Implementácia aplikácie scény v prostredí virtuálnej jaskyne

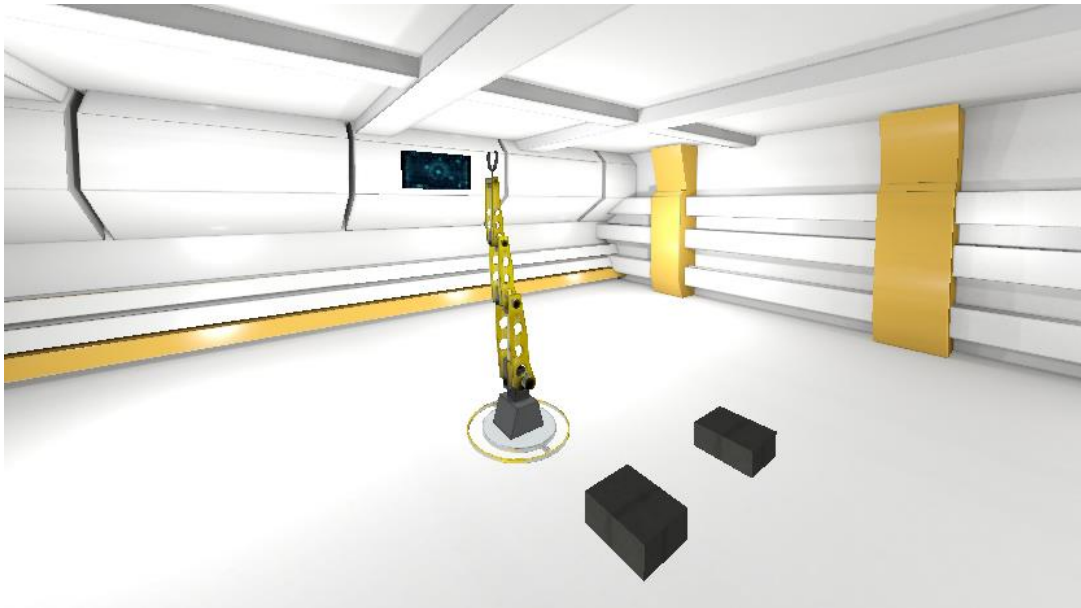
Kapitola zahŕňa realizáciu scény v prostredí virtuálnej jaskyne LIRKIS CAVE. V prvom bode sú definované tri základné scény s grafom a opisom z pohľadu používateľa prostredníctvom funkcionalít a vnemu dosahujúceho pri zobrazení a ovládaní scény v systéme. Druhý bod je orientovaný pre rozšírenie podpory vstupných zariadení a ich správania vo vybraných scénach s porovnaním dostupných prvkov ovládania v scéne, latencií a spôsobom prenášania informácií medzi zariadením a systémom virtuálnej jaskyne. V treťom bode je kapitola zameraná na dosahovanie realistických prvkov scény používaných pri praktickom overovaní diplomového projektu so zameraním sa na sledovanie interakcií medzi používateľom prostredníctvom ovládacích zariadení a prostredím virtuálnej scény.

4.1. Zobrazenie prostredia virtuálneho sveta

Pre systém virtuálnej jaskyne bol vytvorený rozsiahly počet rôznych scén zameraných na testovanie funkcionalít interakcií objektov, transformácií, kombinácie metodík pre zjednodušenie nárokov na systém z pohľadu vykresľovania, dosiahnutie reálnejšieho vnemu v prostredí jaskyne (Príloha A). Zobrazenie scény je v prostredí virtuálnej jaskyne výrazne odlišné oproti verzii používanej pri testovaní komponovanej scény prostredníctvom desktopovej verzie. Hlavným parametrom je zobrazovací uhol kamier a priestoru prostredia v dôsledku používania konštrukcie obrazoviek vytvárajúcich uhol 180 stupňov panoramatického zobrazenia v interakcii s používateľom spolu so zarátaním vertikálneho uhla rovnakej veľkosti. V prípade rôznych scén sa prejavili odlišnosti týkajúce sa virtuálnych prostredí z hľadiska rozmerov a tvarov oproti tvaru konštrukcie virtuálnej jaskyne oproti desktopovej verzii. V prípade opisu rôznych vlastností sú použité scény obsahujúce rôzne metodiky v testovaní vzťahov v trojrozmernom prostredí. Každá zo scén opisuje špecifické vlastnosti s vhodnou realizáciou pre prostredie jaskyne, výhody, nevýhody prípadné nedostatky alebo vylepšenia. Zameranie scén nie je náhodné, dôsledkom použitia špecifického výberu je porovnávanie rozdielov pri aplikovaní transformácií z pohľadu kamery s parametrizáciou pre vzdialenosť kamery od objektov scény alebo vplyvom veľkosti objektov z pohľadu reálneho vnemu.

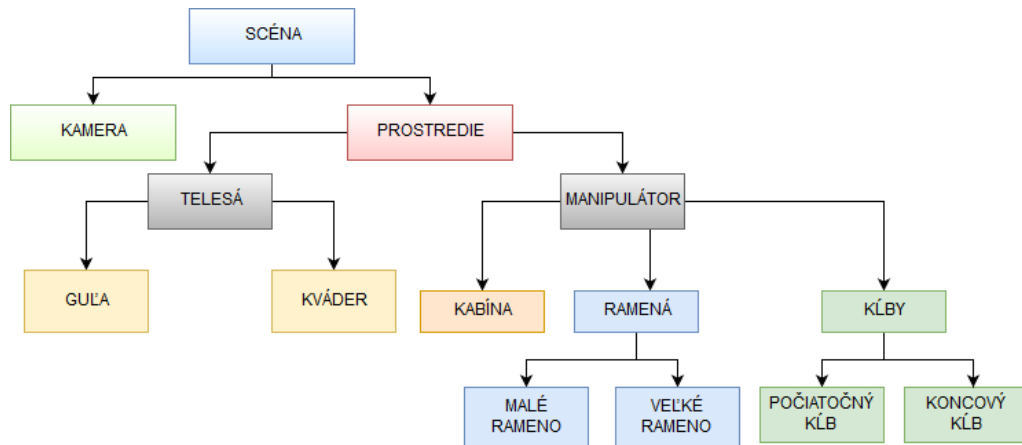
4.1.1. Scéna virtuálneho robotického ramena

Zámerom realizácie scény robotického ramena bolo overenie schopností transformácií trojrozmerných objektov vo virtuálnom prostredí jaskyne. Scéna obsahuje objekty reálneho sveta, nakoľko je potrebné docieľiť v používateľovi vnem ich skutočnej existencie a priblížiť priestor blízky realite. Rameno robota je ovládateľné a polohovateľné podľa potrieb používateľa, umožňuje presúvanie objektov virtuálneho prostredia, podobne presúvanie celej konštrukcie ramena [Obr.19]. V prípade scény je dôležité sledovanie transformácií viazaných k transformačným uzlom objektov a závislostiach prejavujúcich sa pri vykonávaní rotácií alebo pohybu. Rameno obsahuje štyri kĺby reprezentované transformačnými uzlami zabezpečujúcimi horizontálne rotácie a koncový úchyt rotujúci v horizontálnom a vertikálnom smere pre umožnenie práce s ostatnými objektmi scény. V scéne bolo potrebné uvážiť možnosti pre zabezpečenie kolízií, nakoľko samotné rameno vytvára interakcie s ostatnými objektmi scény a priestorom miestnosti, v ktorom je obsiahnuté. Dôležitým bodom realizácie scény je ohraničenie priestoru v podobe miestnosti s obmedzením prenikania objektov mimo virtuálneho sveta.



Obr. 19 Scéna s robotickým ramenom.

Príprava scény zahŕňala implementáciu základných vlastností pre pohyb a rotácie, kolízie. Z pohľadu používateľa bol rovnako aplikovaný OptiTrack so snímaním pohybu a prispôbením pohľadu používateľa v prostredí virtuálnej jaskyne. V prípade realizácie prostredia bolo potrebné zohľadniť obmedzenia systému SuperEngine, ktorý absentuje v možnostiach využitia knižníc s implementovanou fyzikou pre virtuálne prostredie, rovnako aplikácie svetelných efektov. V prípade grafu scény kinematického ramena so zahrnutými dynamickými vlastnosťami je potrebné uviesť spôsob používania kinematiky. Rameno obsahuje časti kĺbov a menších ramien slúžiacich pre docielenie náklonov a rotácií v priestore. Dôležitá je predovšetkým väzba medzi jednotlivými objektmi ramena, s využitím počiatočného a finálneho bodu rotácie. Rameno obsahuje základňu tvoriacu počiatočný bod s postupnými dopĺňaním rôznych častí kĺbov a ďalších čiastkových ramien vykonávajúcich funkciu pohybu a rotácie. Viazanie grafu scény je postupné od počiatočnej základne po posledný objekt ramena [Obr.20]. Prostredníctvom ramena je možné zachytávať rôzne druhy dynamických objektov reprezentovaných v trojrozmernom priestore s možnosťou manipulácie transformácií posunu s využitím priamej kinematiky smerujúcej od počiatočného bodu viazania kĺbu po koncový bod záchytnej časti ramena.



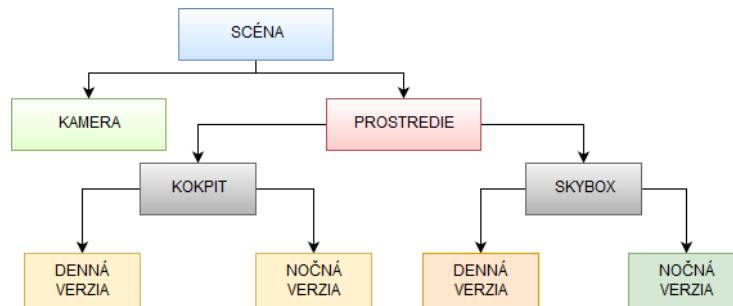
Obr. 20 Graf scény kinematického ramena.

Celkovo sa preukázala neschopnosť viazania uzlov scény a objektov prostredníctvom pripojenia k niektorého z objektov k nadradenému uzlu scény. Pripojenie uzlov v prípade ramena a prenášaného objektu viaže všetky použité transformácie s ostatnými pod-uzlami ramena, ktoré spôsobujú komplikácie pri implementácii základnej logiky presúvania objektov. Spomínaný problém je cítiteľný z pohľadu vizuálnej stránky. Metóda určená pre priradenie objektu k nadradenému uzlu alebo objektu k nadradenému objektu aplikuje všetky predošlé transformácie predošlej štruktúry na novo pripojený objekt. V takomto prípade je náročné vytváranie transformácií ramena, viazaných na pripojený objekt pri prenášaní ostatných objektov. Metóda pre odpojenie objektu od transformačného uzla v systéme SuperEngine nereaguje. Vyššie časové nároky boli pri realizácii výrazné v prípade používania svetelných prvkov pre dosahovanie vnemu existencie reálneho svetla v prostredí a generovaní tieňov podľa pohybu objektov v miestnosti. Používaním viacerých svetelných prvkov výrazne narastala náročnosť pri vykresľovaní scény, správanie objektov prejavovalo mierne latencie, zaostávanie v reakcií podľa potrieb používateľa. Celkový vizuálny dojem je primeraný náročnosti scény, funkcionality je výrazne obmedzená v dôsledku absencií riadenia scény aplikovaním viazania transformačných uzlov ramena.

4.1.2. Scéna kokpitu lietadla

Dôvodom realizácie priestoru kokpitu lietadla bola myšlienka docieľiť virtuálny priestor ohraničený priestorom virtuálnej jaskyne. Virtuálny priestor kokpitu je veľkosťou identický skutočnému fyzickému priestoru virtuálnej jaskyne. Používanie takejto scény dosahuje efekt zameraný na vnem používateľa s rozpoznávaním vzdialeností blízkych objektov kokpitu a prostredia scény s obsahom krajiny. Podobne aplikuje transformácie kamery pri pohybe používateľa v systéme jaskyne s využitím zariadenia OptiTrack. Hlavnou úlohou scény nie je priama simulácia lietadla ale využívanie možností pre zobrazenie objektu s pocitom reálneho vnemu, ako keby

sa používateľ nachádzal v priestore skutočného kokpitu. Graf scény [Obr.21] zobrazuje jednotlivé komponenty v hierarchii objektov viazaných v trojrozmernom prostredí bez výraznej členitosti. Dôvodom je testovanie scény z pohľadu vnemu používateľa pri dosahovaní efektov blízkych reálnemu prostrediu z grafického pohľadu.



Obr. 21 Graf scény kokpitu lietadla.

Scéna aplikuje skybox pre dosiahnutie reálnejšieho zobrazenia sveta v trojrozmernom prostredí, reprezentovaný 360 stupňovou fotografiou opisujúcou panoramatický záber krajiny. Testovaním používania skybox okolia je efekt v dosahovaní reálnejšieho vnemu oveľa presvedčivý oproti klasickému vytváraniu objektov krajiny. Problémom zvolenej scény bolo riešenie jednoduchšej dynamiky prostredia, pre vytváranie pocitu pohybu lietadla v priestore. Prvotné riešenie predstavovalo použitie hĺbkovej mapy sveta aplikovanej na povrch objektu gule, ktorá by reprezentovala terén. Nakoľko je v prípade dynamickej scény v SuperEngine náročné používať generátory terénu s výberom rôznych objektov, scéna uplatňuje používanie dvoch objektov reprezentujúcich svet a oblohu v prostredí. Generovanie terénu nepredstavuje zložité riešenie pre logický návrh vytvárania a dodávania objektov, vysoká náročnosť vzniká pri ich transformácií v teréne, ktorá sa môže vyznačovať latenciami v podobe oneskorených reakcií skriptu a systému virtuálnej jaskyne. V prostredí, ktorého úlohou je vzbudzovanie reálneho vnemu priestoru pre používateľa dôsledné, je nevhodné realizovať zložitosť spôsobujúcu latencie. V prípade zobrazenia kokpitu bol použitý objekt s nanášaním generovaných textúr aplikujúcich tieň a svetelné efekty [Obr.22]. Dôvodom pre vytváranie textúr prostredníctvom modelovacieho nástroja miesto skriptovania svetelnej scény bola rovnako splnená požiadavka zachovania reálneho dojmu a zamedzenie latencií systému. SuperEngine prejavuje pri aplikácii svetiel dodávaných interne prostredníctvom skriptu, a ich volaním priamo v jadre výrazné latencie. Latencie sa zvyšujú pri náraste náročnosti scény, zložitosti používaných objektov a ich štruktúry, pričom samotný engine neposkytuje generovanie svetelných lúčov, ani samotného zdroja svetelného prvku. Výsledná scéna pôsobí presvedčivo, reakcie systému dosahujú úroveň bez vytvárania latencií, štruktúry a výsledné aplikované modely vzbudzujú dojem návštevy

reálneho kokpitu. Z pohľadu prostredia sú statické skyboxy vhodné, vytvárajú dojem skutočného prostredia v okolí, dynamické prostredie vrátane terénu a oblohy je výrazne odlišné, predovšetkým umožňuje demonštráciu rotácií sveta v priestore.

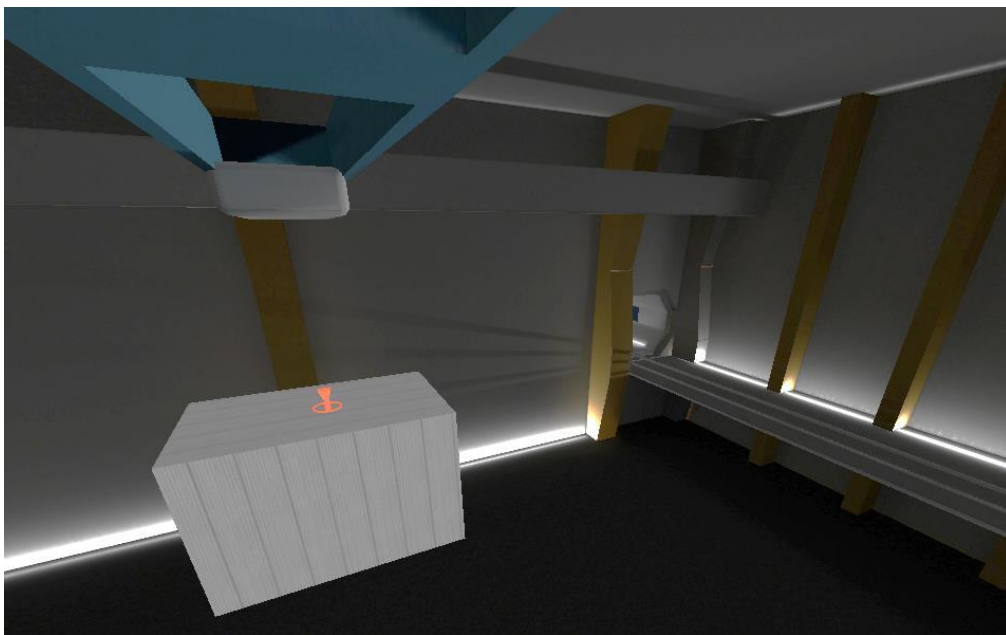


Obr. 22 Model kokpitu lietadla Boeing 737 v systéme virtuálnej jaskyne.

Existujú typy scén, pre ktoré je dôležité transformovať model v priestore vykonávajúci pohyb miesto transformácií pohybu a rotácií okolitého prostredia. V prípade riešenia scény kokpitu v interakcii s prostredím sveta a aplikovaní objektu gule reprezentujúcej terén bolo najvhodnejším spôsobom aplikovať transformácie na prostredie sveta. Dôvodom je zložitosť interpolácií v prípade ak by boli transformácie viazané na model kokpitu, nakoľko by bolo náročné riešenie pohybu modelu kokpitu okolo gule, terénu. Rotovaním samotnej gule s terénom je použitý jeden druh rotácií v smere jednej osi, miesto použitia interpolačných metód pohybu a rotácie kamery s kokpitom a zbytočného zvyšovania náročnosti pre výpočtový systém. Rotáciou sveta je dojem z priestoru kokpitu rovnaký, rozdielna je metodika aplikácie použitia transformačných metód v trojrozmernom priestore a ich viazaní na zvolené objekty. Scéna rovnako aplikuje možnosti používania rozhrania prostredníctvom ovládacích prvkov JavaConsole s využitím implementácie tlačidiel dostupných v používateľskom rozhraní obrazovky. Tlačidlá umožňujú prepínať jednotlivé prostredia s voľbou dynamickej alebo statickej scény, dňom alebo nocou a aplikáciou skybox prvkov.

4.1.3. Scéna halového žeriavu

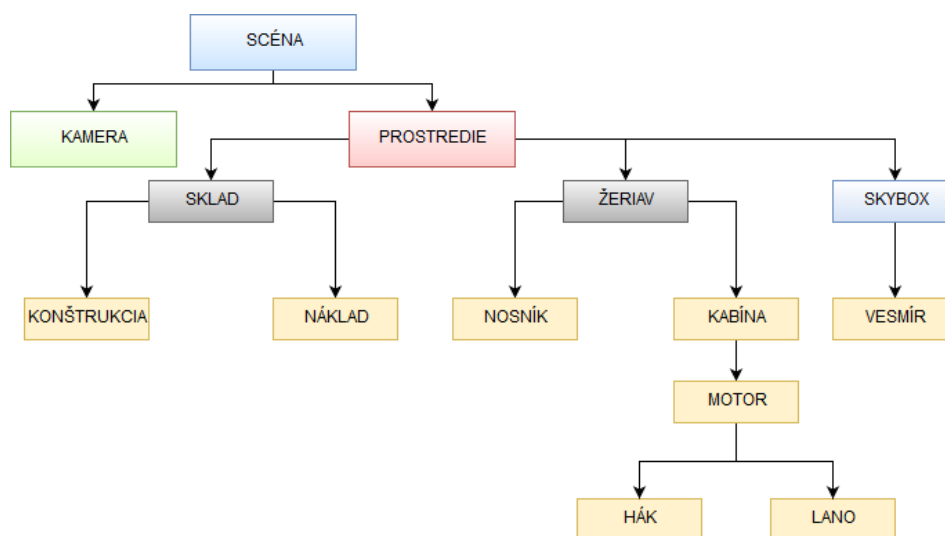
V prípade scény halového žeriavu bolo potrebné overiť vhodnosť reakcií systému na vstupy podávané prostredníctvom vstupného zariadenia a používateľského rozhrania dostupného zo strany Java Console. Zvolené bolo používanie zariadenia OptiTrack pre reálnejší dojem a interaktivitu medzi používateľom a systémom virtuálnej jaskyne. Scéna halového žeriavu aplikuje niekoľko pohľadov kamery s voliteľným výberom prostredníctvom interaktívnych tlačidiel. Nevýhodou je absencia implementácie viacerých pohľadov kamier naraz, čo znemožňuje používanie spätných zrkadiel alebo zobrazením viacerých kamier v jednom pohľade. Absenciou zobrazenia pohľadov z viacerých kamier v reálnom čase rovnako ovplyvňuje problém implementovať aktívne odrazy plôch objektov. Prostredie žeriavu obsahuje generované textúry s využitými svetelnými prvkami ako výstupmi modelovacieho nástroja [Obr.23]. Dôvodom je spomínaná absencia vlastností svetelných prvkov spomínaná v predošlej podkapitole.



Obr. 23 Halový žeriav v SuperEngine.

Úlohou používateľa v scéne halového žeriavu je variabilné prenášanie objektov nákladu v prostredí virtuálneho sveta. Konštrukcia žeriavu využíva prenášajúci nástroj reprezentujúci magnet s úchytom pre priťahovanie objektov s využívaním implementácie interakcií pre porovnanie polohy magnetu a objektu nákladu. Pôvodným zámerom implementácie bolo obsiahnutie ovládacieho prvku Joystick pre zabezpečenie riadenia pohybu kabíny a nákladu. Pre príkladný opis je použitý graf scény s aplikáciou žeriavu prenášajúceho náklad v prostredí virtuálnej jaskyne. Scéna pozostáva z dvoch základných častí, žeriavu a miestnosti skladu

s delením na jednotlivé druhy objektov. Sklad je opísaný objektmi [Obr.24] konštrukcie pre zobrazenie pevných statických častí scény a nákladu pre presúvanie na rôzne miesta scény prostredníctvom použitia žeriavu. Samotný žeriav obsahuje niekoľko základných častí, nosník, kabínu, motor, lano a hák. Nosník je charakteristický pre konštrukčnú časť, ktorá je v skutočnom prostredí viazaná s prvkami kabíny, v prípade implementácie bolo potrebné zabezpečiť aby nosník a kabína dosahovali len niektoré z väzieb, pretože v opačnom prípade by viazanosť kabíny bola statická bež možnosti voľby variabilného pohybu na konštrukcii nosníka. Kabína sa takto môže pohybovať v smere do ľavej a pravej strany bez toho, aby takýto pohyb opisoval rovnako nosník žeriavovej konštrukcie. Ďalším podradeným objektom kabíny je motor, viazaný vo všetkých transformačných vlastnostiach. Samotné lano a hák sú väzbami rozdielne, problémom je možnosť škálovania objektu. V prípade metódy škálovania v prostredí SuperEngine sú rovnako škálované aj objekty nachádzajúce sa hierarchicky nižšie, preto by nebolo vhodné v prípade rozťahovania lana pri spúšťaní rovnako škálovať hák. Ošetrovanie pre rozdelenie väzieb v závislosti od jedného bodu scény umožňuje aplikovať transformácie v rámci každého objektu zvlášť. V prípade škálovania objektov od seba nezávisle, u ktorých je potrebné zabezpečiť niektorý z druhov väzieb, je dôležité naviazať objekty na bod alebo objekt nerealizujúci transformácie s deformačným prejavom. V prípade využívania zložitej scény s obsahom rôznych väzieb alebo zložitej hierarchie objektov v scéne je jednoduchšie zabezpečiť funkcionality implementáciou prostredníctvom návrhu metód, ktoré môžu realizovať interakcie medzi viacerými objektmi naraz. Výhodou je zjednodušenie náročnosti implementácie a univerzálnosť metódy pre zabezpečenie správnej funkcionality a väzieb objektov trojrozmernej scény.



Obr. 24 Graf scény žeriavu.

Výsledná scéna obsahuje základnú implementáciu správania halového žeriavu so zvukovými efektmi pri využívaní pohybu žeriavu, lana alebo pohybu nákladu v priestore. Realizácia ovládacieho zariadenia bola nahradená ovládacím prvkom klávesnice a myši z dôvodu absencie knižníc dovoľujúcich variabilné pripájanie akýchkoľvek ovládacích zariadení. Neskôr bola obohatená o používanie zariadenia Joystick, podporovaného externou aplikáciou vytvorenou pre doplnenie modularity systému a rozšírenie podporu vstupných zariadení. Zariadenie Joystick umožňuje používateľovi vykonávanie plynulého pohybu v prostredí virtuálnej scény, dovoľuje používateľovi narábať s nákladom prostredníctvom tlačidiel, ovládať motor zdvihu a nastavovať rýchlosť pohybu kolesa motora alebo kabíny žeriavu. Používanie zariadenia obohacuje systém jaskyne o reálnejší vnem z pohľadu používateľa miesto aplikovania ovládacích prvkov pomocou klávesnice alebo myši. Dosahovaním prvkov ovládania pripomínajúcich prvky blízke reálnemu žeriavu je celkový dojem viac presvedčivejší, reálnejší.

4.2. Využitie vstupov systému

Implementácia skriptov pre prostredie SuperEngine, softvér aplikovaný vo virtuálnej jaskyni umožňuje používanie vstupných zariadení obsahujúcich vstupy tlačidiel klávesnice, vstupy údajov pre myš, prenášanie vstupov prostredníctvom zariadenia 3D mouse a interaktívne sledovanie pohybu bodov pomocou zariadenia OptiTrack. Každý neštandardný vstup potrebný pre aplikáciu v systéme je nutné implementovať na modulárnej úrovni s využitím skriptovacieho jazyka Ruby 1.8.6. Uvedená verzia jazyka Ruby spôsobuje komplikácie pri implementácii nových zariadení v dôsledku neaktuálnosti oproti dnešným vstupným zariadeniam z pohľadu komunikácie. Vstupy realizované využívaním prípojných zariadení formou klávesnice a myši boli v predošlom stave pred vykonávaním úprav diplomového projektu nedostupné, sprístupnili sa postupne pri aktualizovaní verzie s možnosťou rozšírenia vstupných zariadení a používateľských rozhraní. Každé zo vstupných zariadení je potrebné konfigurovať v systéme pre dosiahnutie bezchybnej komunikácie a rovnako zabráneniu nechcených vstupov počas vykonávania scény. V prípade konfigurácie vstupných zariadení existujú preddefinované konfiguračné súbory, ktoré sú definované priamo zo strany spoločnosti Slovakia SuperComputers a je možné ich variabilne meniť podľa potrieb používateľa. Využívanie vstupov dovoľuje aplikáciu viacerých vstupných zariadení v reálnom čase, ktoré predstavuje výhodu v prípade ovládania objektov komplexnejších scén alebo funkcií v jednom prostredí. V prípade nesprávne nakonfigurovaného ovládacieho zariadenia alebo jeho nesprávneho volania v skripte scény sú reakcie systému virtuálnej jaskyne pri spustení prostredia nefunkčné. Dôvodom je volanie zariadenia, ktoré nie je definované a nakonfigurované, pričom nie je možné aplikovať očakávané vstupy. Takáto chyba často spôsobuje pozastavenie systému a zablokovanie scény prostredia, SuperEngine absentuje dostupnosťou

chybových hlásení a informácií alebo upozornení používateľa systému pri volaní nesprávne konfigurovaných ovládacích zariadení.

4.2.1. Overenie vstupov pre klávesnicu

Podpora zariadení v systéme virtuálnej jaskyne bola v predošlom stave striktno obmedzená pre zariadenia priamo implementované spoločnosťou Slovakia SuperComputers, ktoré nezahŕňali vstupy zariadenia klávesnice. Podpora klávesnice bola aktualizovaná počas realizácie diplomového projektu dodaním funkcionality s konfiguráciou pre čítanie znakov ASCII priamo v implementácii skriptov scén. Samotný systém SuperEngine podporuje vstupy z klávesnice po vykonaní konfigurácie vstupných zariadení a dosadením aktualizovaného používateľského rozhrania, ktoré poskytla priamo spoločnosť Slovakia SuperComputers pre umožnenie rozšírenia vstupných zariadení. Potrebná bola rozsiahla zmena konfiguračných súborov zariadení obsiahnutá v module ControlCenter. Vstupy z klávesnice sú riadené prostredníctvom konfiguračného súboru obsiahnutého v ceste súborovej štruktúry modulu ControlCenter na hlavnom počítači riadiaceho komunikáciu jaskyne - Master. Funkcia konfiguračného súboru spĺňa dôležitú úlohu pri inicializácii vstupných zariadení načítavaných pri zapnutí systému SuperEngine. Štruktúra konfiguračného súboru definuje zariadenie, typ, názov a cestu v rámci adresy, ktorá zodpovedá adrese zariadenia v sieti zasielajúceho vstupy priamo do modulu ControlCenter. Modul ControlCenter prekladá vstupy získavané zo zariadenia posielajúceho vstupné signály, podľa druhu zariadenia, špecifikácie a adresy. V ďalšom kroku definuje typ signálu a obsah signálu. Rozpoznaný signál ďalej porovnáva so vstupným skriptom a vyhľadáva zhody v implementácii so vstupným signálom. V prípade zhody vykonáva preklad vstupu s viazaním na implementovanú funkcionality a zasiela pokyny pre vizualizačné jadro. Úlohou vizualizačného jadra je podľa pokynov vykonávať aktivitu v trojrozmernom prostredí virtuálneho sveta a vykresľovať scénu na zobrazovacích zariadeniach virtuálnej jaskyne. Nevýhodou z pohľadu SuperEngine je absencia sledovania vstupných zariadení počítačov mimo výpočtovej jednotky master. V opačnom prípade systém nedokáže pracovať korektne, pričom nedokáže spracovávať signály vyplývajúce zo vstupov zariadení pripojených na ostatných počítačoch a podávať ich formou výstupov pre vstupy vizualizačného jadra, napriek pripojeniu všetkých zariadení v rovnakej lokálnej sieti. Problém je čiastočne riešiteľný používaním aplikácií pre VRPN server, s možnosťou viazania komunikácie prostredníctvom akéhokoľvek iného počítača. Nevýhodou je problém vznikajúci pri obmedzeniach bránou firewall a podobne používaní viacerých zariadení v reálnom čase, nakoľko SuperEngine v takomto prípade nedokáže spracovávať prúd dát súčasne kvôli problému s obsadením portov.

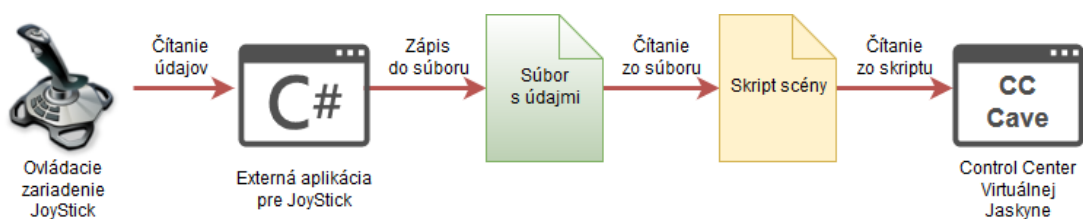
4.2.2. Overenie vstupov pre myš

Podpora vstupného zariadenia obsahujúceho myš bola podobne nedostupná v konfigurácii systému. Potrebné bolo vykonať zmeny konfigurácií vstupných zariadení s doplnením rozhrania pre prídavné zariadenie myš. V prípade klávesnice a myši boli zariadenia v systéme implementované pri vývoji systému SyperEngine, ich absencia závisela od konfigurácie vstupov. Myš obsahuje vstupy plynulejšieho charakteru, opačné oproti vstupom viazaným na udalosti vstupov klávesnice. Vstupy realizované zariadením myš nesledujú priamo vstupné hodnoty z hardvérového prvku, ktorý reprezentuje myš samotnú, realizujú načítavanie hodnôt pomocou pozície umiestnenia kurzora na obrazovke. Riešenie značne pôsobí triviálne, problémom je samotné rozlíšenie zobrazovacieho zariadenia, v prípade jaskyne sa načítavanie pozície zariadenia myš viaže k rozlíšeniu zobrazovacích zariadení pripojených k počítaču označeného master. Predstavme si situáciu s používaním vstupného zariadenia myš pri rozlíšení obrazovky 1200 pixelov na 800 pixelov. Hodnota parametra sledujúceho horizontálnu pozíciu myšky je obsiahnutá v intervale $\langle 0,1200 \rangle$ a hodnota parametra sledujúceho vertikálnu polohu je obsiahnutá v intervale $\langle 0,800 \rangle$. Skript scény bude vo zvolenom prípade prenášať na vstupe hodnoty myši z definovaných intervalov. Pri vykonaní zmeny rozlíšenia zobrazovacieho zariadenia je nutné aktualizovať skript scény, nakoľko sa môže očakávať zmena správania funkcionality scény viazanej na hodnoty podávaných zo strany vstupného zariadenia myš. Takáto záležitosť predstavuje komplexný charakter, často nevhodný pri implementácii s opakovanými zmenami rozlíšenia scény, pričom pre každú zmenu zobrazovacieho rozlíšenia je nutné upraviť skript scény so sledovaním hodnôt pozície zariadenia myš. Rovnakým problémom pre sledovanie vyhradených intervalov je zamedzenie pohybu bez obmedzení, myš nedokáže vykonávať rotácie v akomkoľvek smere nepretržite. Hranice intervalov rozlíšenia spôsobujú zastavenie kurzora a obmedzujú tým aplikáciu zariadenia pre prípad nekonečnej rotácie podľa intenzity pohybu. Príkladné je porovnanie herných aplikácií s možnosťou otáčania používateľa v scéne s nekonečnou panoramatickou rotáciou. Používateľ pomocou zariadenia myš pohybuje do strán bez obmedzení zastavenia v scéne, v prípade engine vie používateľ pohybovať alebo rotovať scénou len v obmedzenom intervale rozlíšenia monitora.

4.2.3. Overenie vstupov pre Joystick

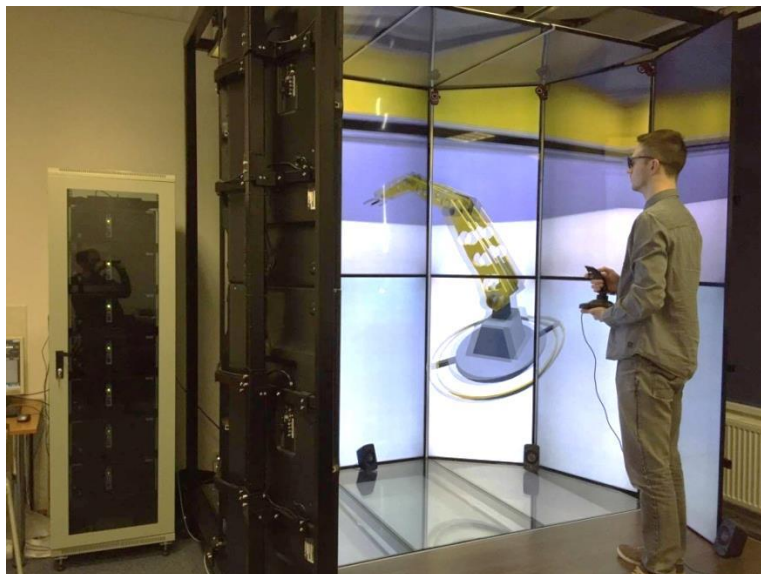
Prvotnou myšlienkou pri realizovaní diplomového projektu bolo rozšírenie podpory vstupov prostredníctvom zariadenia Joystick [34]. V dôsledku zachovania platformity a kompatibility pre pripojenie štandardných zariadení Joystick bolo aplikované načítavanie informácií zo zariadenia s využitím externej aplikácie implementovanej v jazyku C Sharp (C#) a knižnic SharpDX [35]. Realizovaná aplikácia odchyťava údaje zo zariadenia a zapisuje jednotlivé

informácie do súborov umiestnených na sieťovom disku výpočtovej jednotky Master v systéme virtuálnej jaskyne [Obr.25]. Skript scény virtuálnej jaskyne sleduje zmeny vykonávané v súboroch s informáciami o stave jednotlivých parametrov pripojeného zariadenia a prostredníctvom implementovaných funkcií prenáša údaje do prostredia scény. V prípade načítavania údajov zo zariadenia Joystick bolo realizované sledovanie polôh osi X a Y s pripojením 6 tlačidiel obsiahnutých na štandardnom zariadení. Dôvodom pre sledovanie uvedeného počtu tlačidiel je bežný rozsah vstupov herných zariadení, vedený ako štandard. Rovnako je možné rozsah meniť, v prípade ak Joystick obsahuje viacero vstupov tlačidiel. Základné rozhranie aplikácie umožňuje sledovať pripojené zariadenie, čítanie údajov a zápis na sieťový disk bez vplyvu prerušenia načítanej scény alebo blokovania systému jaskyne.



Obr. 25 Zobrazenie komunikácie externej aplikácie určenej pre zariadenie Joystick.

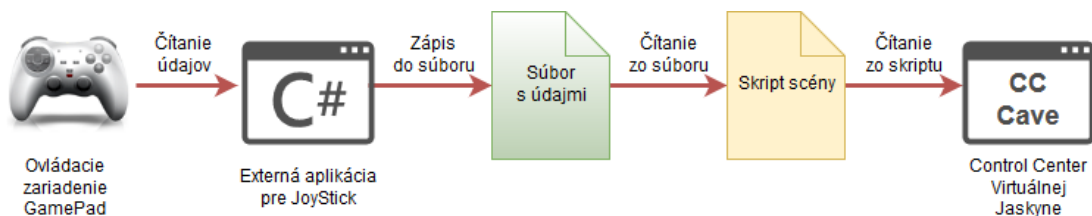
Doplnenie platformy vstupných zariadení o Joystick rozšírilo využitie systému virtuálnej jaskyne a ponúklo plynulé ovládanie scénických objektov [Obr.26]. Prínosom využitia externej aplikácie je pripojenie ovládacieho zariadenia na ktoromkoľvek výpočtovom zariadení virtuálnej jaskyne s využitím prenosu údajov pomocou siete. V prípade používania zariadenia Joystick pomocou externej aplikácie neboli prejavené žiadne latencie v scénach, rovnako v systéme virtuálnej jaskyne. Aplikovanie jednoduchého zápisu na sieťový disk výpočtovej jednotky Master umožňuje používateľovi zobraziť prehľad v každom súbore s hodnotou údajov zariadenia Joystick. Dátový tok v prípade zápisu súborov na sieťový disk nevytvára spomalenie komunikácie, objem prenesených dát medzi aplikáciou, sieťovým diskom a skriptom scény je nízky v dôsledku zápisu krátkych číselných reťazcov. Externá aplikácia dosahuje vysokú presnosť, neovplyvňuje systém z pohľadu vnútornej štruktúry, v dôsledku ktorej nevytvára zablokovanie chodu alebo neočakávané reakcie systému. Výhodou externých aplikácií implementovaných pre používanie ovládacích zariadení je ich nezávislosť od softvéru SuperEngine. Zariadenia je možné pripájať a odpájať počas behu scény, bez pádu systému alebo odopierania niektorej funkcionality. Rovnakým spôsobom je riešiteľné akékoľvek zariadenie pri používaní uvedenej komunikácie aplikácie so skriptom scény. V prípade trvalého výpadku zariadenia je možné pokračovať v simulácii scény bez jeho používania.



Obr. 26 Používanie zariadenia Joystick v systéme virtuálnej jaskyne.

4.2.4. Overenie vstupov pre Gamepad

Vytvorením externej aplikácie pre používanie zariadenia Joystick bola realizovaná myšlienka rozšírenia systému virtuálnej jaskyne o ďalší z prvkov vstupných zariadení. Dôvodom výberu zariadenia Gamepad bolo využívanie viacerých osí pre simuláciu prostredí so zložitejšími transformáciami v trojrozmernom priestore. Gamepad obsahuje štyri osi, simuláciu pohybov v smere osí X,Y,Z a rotáciu okolo osi Z, ktoré je možné aplikovať nezávisle od seba a ovládať komplexnejší objekt alebo scénu. V prípade zabezpečenia platformity s rôznym Gamepad ovládačom systém umožňuje okrem využívania osí aplikovať 6 štandardných tlačidiel s možnosťou neobmedzeného rozšírenia ich počtu. Úroveň komunikácie zariadenia a systému je rovnaká ako v prípade používania ovládacieho prvku Joystick [Obr.27]. Zariadenie komunikuje s externou aplikáciou implementovanou v jazyku CSharp prostredníctvom knižníc SharpDX, ktorá výstupy zariadenia zapisuje do súborov sieťového disku systému virtuálnej jaskyne. Skript scény sleduje stav hodnôt parametrov súborov vstupného zariadenia na sieťovom disku a prenáša ich do trojrozmerného prostredia podľa predurčenej funkcionality alebo využitia.



Obr. 27 Zobrazenie komunikácie externej aplikácie určenej pre zariadenie Gamepad.

Využívaním zariadenia neboli preukázané latencie alebo výpadky systému. Využívanie externej aplikácie nezasahuje do vnútornej štruktúry systému, pričom nespôsobuje zásahy v komunikácií

medzi jednotlivými modulmi SuperEngine. Výsledkom je zabránenie spomalenia systému, neočakávaných reakcií alebo správania modulov. Externú aplikáciu je možné používať na akomkoľvek zariadení s operačným systémom Windows 7 a vyššie v prostredí virtuálnej jaskyne po vykonaní konfigurácie sieťového pripojenia. Aplikovaním rozšírenia systému virtuálnej jaskyne o zariadenie Gamepad je možné docíliť plynulejšie ovládanie scénických objektov, rozšírenie funkcionalít systému, riadenie komplexnejších scén. Výhodou zariadenia je rovnako intuitívne používanie a jednoduchosť pri ovládaní objektov scény [Obr.28]. Používateľ môže v reálnom čase ovládať objekt podľa intenzity posúvania osí vbudovaných dvoch rotačných ovládačov alebo využívať tlačidlá pre vykonávanie udalostí na scéne. Gamepad umožňuje prepínanie do stavu analógového a digitálneho riadenia signálov, pričom je možný voliteľný spôsob sledovania intenzity ovládania rotačných ovládačov. V analógovom móde sú rotačné ovládače viazané na jemný pohyb snímaný intenzitou vychýlenia každej páky ovládača alebo v digitálnom móde snímajúcom diskretný stav bez sledovania intenzity.



Obr. 28 Používanie zariadenia Gamepad v systéme virtuálnej jaskyne.

4.2.5. Overenie vstupov pre OptiTrack

Používanie systému OptiTrack obohacuje systém virtuálnej jaskyne o moderný prvok zariadení a inováciu v interakcií používateľa s virtuálnym svetom. Pomocou značiek umiestnených na hlave používateľa a triangulácie s využívaním infračervených kamier rozmiestnených v systéme virtuálnej jaskyne je možné docíliť prispôsobenie scény pre dosiahnutie reálneho vnemu. V systéme je obsiahnutých 8 kamier sledujúcich panoramatické a vertikálne zmeny polohy každého zo snímaných bodov, pomocou ktorých sa dokáže scéna prispôbovať používateľovi podľa jeho polohy. Rozdiely pri triangulácii preukazujú výrazné odchýlky, nie sú ale považované

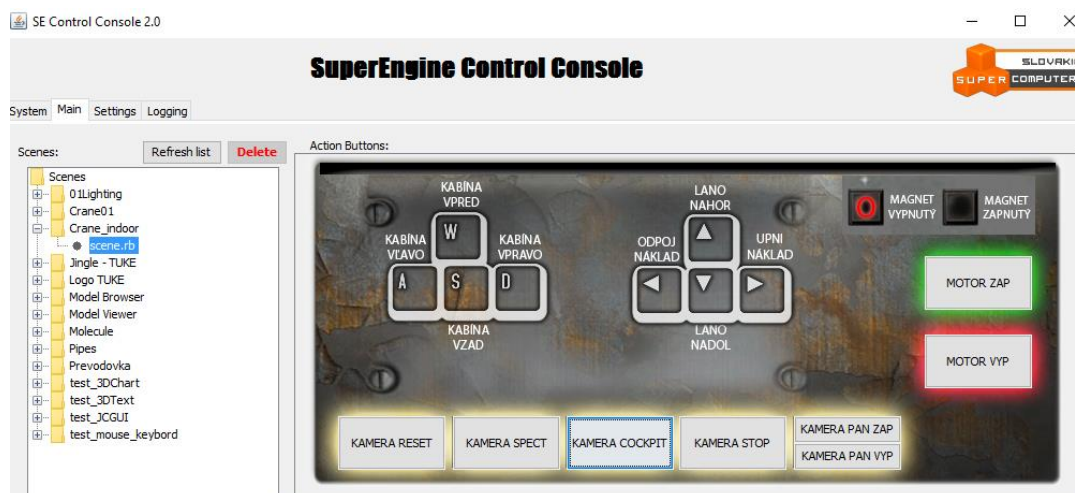
za chyby samotného trackovania – sledovania bodov kamerami v priestore. Najčastejší rozdiel je preukázaný pri používaní scén s blízkou a ďalekou vzdialenosťou rozmiestnených objektov. Objekty nachádzajúce sa v dostupnej vzdialenosti k používateľovi, napríklad kokpit lietadla, kabína žerjavu alebo akýkoľvek objekt s pozíciou blízkej ohraničenie systému jaskyne je zobrazovaný s použitím zariadení OptiTrack korektne. Korektným zobrazením rozumieme zmenu veľkosti objektu v závislosti zmeny polohy alebo vzdialenosti používateľa v systéme. Blízke objekty sú pri vzdialení používateľa od obrazoviek systému automaticky zmenšené, logicky sa mení veľkosť objektu ak sa používateľ vzdiali. Nekorektné zobrazenie je často sprevádzané v scénach s objektmi vzdialenými mimo hraníc rozmerov systému virtuálnej jaskyne. V takomto prípade nastáva skreslenie obrazu, deformácii objektu a uhla zobrazenia. Riešenie predstavuje posúvanie kamery a postupné približovanie sa k objektom pri ich demonštrácií, výsledný efekt nesprávneho prispôbenia obrazu nastáva len v prípade ak sú objekty vzdialené príliš ďaleko, alebo sú rozmerom niekoľko násobne väčšie oproti rozmerom virtuálnej jaskyne [Obr.29] Dôsledné je pripomenúť využívanie stereoskopie s používaním prispôbenia obrazu s výskytom anomálií spôsobujúcich spomínané deformácie. Stereoskopický obraz sa pri zmene vzdialenosti používateľa od konštrukcie jaskyne mení, prispôbuje sa zobrazovanie scény pre ľavé a pravé oko. Častou komplikáciou je zorný uhol zobrazenia scény, prispôbovaný pri zmene vzdialenosti používateľa v systéme, často deformovaný pri stereoskopickom zobrazení. Príčinu predstavuje nevhodné implementovanie v systéme SuperEngine, ktoré axnomoetricky do istej vzdialenosti poskytuje korektné zobrazenie trojrozmerného priestoru. Rozdiely sa prejavujú nielen pri vystúpení používateľa z prostredia virtuálnej jaskyne, rovnako sú viditeľné v prípade jeho prítomnosti a zmeny polohy v blízkej vzdialenosti od obrazoviek. Takáto anomália je viditeľná v scénach na objektoch, ktorých rozmery alebo vzdialenosť sú niekoľkonásobne väčšie od rozmerov prostredia jaskyne.



Obr. 29 Rozdiely dosahované pri používaní systému OptiTrack.

4.2.6. Aplikovanie interaktívnych tlačidiel

Využívanie vstupov systému poskytuje vytváranie variabilného používateľského rozhrania zobrazovaného prostredníctvom modulu JavaConsole. Tlačidlá v pôvodnej verzii modulu JavaConsole nedokázali spolupracovať so systémom virtuálnej jaskyne korektné, funkcionality sa prejavovala častým zastavením scény a neschopnosťou reakcie všetkých prípojných vstupných zariadení. Problém bol čiastočne riešiteľný reštartovaním systému celého klastra, opačne pri reštartovaní len vizualizačného jadra sa problém opakoval. Riešením bola aplikovaná aktualizovaná verzia JavaConsole s dostupným rozhraním pre riadenie vstupov pomocou aplikácie realizujúcej VRPN komunikáciu. Tlačidlá je možné variabilne prispôbovať v skripte scény, viazať na rôzny druh funkcionalít, prispôbovať rozmerom podľa používateľa alebo druhu scény. Rovnako je možné vytváranie kariet, vstupných textových polí alebo elementov so zobrazením percentuálneho stavu niektorej z premenných obsiahnutých v skripte scény. Množina prvkov pre vytváranie používateľského rozhrania disponuje vytváraním zoznamov so zahrnutím rôzneho výberu a kombinácie vstupných hodnôt, prípadne výstupov údajov v okne konzoly [Obr.30]. Výhodou používateľského rozhrania je variabilita, prispôbivosť pre potreby používateľa, jednoduchosť a použiteľnosť. Záporné vlastnosti sú z pohľadu používateľa len triviálne, tlačidlá je možné viazať na kliknutie, nie je možné ich pridržať alebo ponechať stlačené. Každá funkcionality tlačidiel a elementov je kombinovateľná, predstavuje využitie v aplikáciách pre prácu s objektmi virtuálneho sveta alebo zadávaním parametrov scény. Používateľské rozhranie umožňuje zobrazenie parametrov scény, ktoré môžu zodpovedať číselným údajom alebo reťazcom znakov. JavaConsole je rovnako aplikovateľná na zariadení s dotykovým displejom a operačným systémom Windows. V súhrnnom pohľade je použiteľná pri akomkoľvek type scény, s možnosťou výberu vlastných funkcionalít podľa druhu potreby.



Obr. 30 Interaktívne tlačidlá scény halového žeriavu.

4.3. Dosiahnutie reálnejšieho vnemu

Dosahovanie reálnejšieho vnemu v prostredí virtuálnych jaskýň je moderným trendom v zobrazovaní trojrozmernej scény. Časté rozširovanie metodík v oblasti aplikácie svetelných efektov, vlastností materiálov, rovnako dynamiky prostredí a spätnej väzby rozširuje uplatnenie systémov virtuálnych jaskýň a otvára ďalšie možnosti pre rôzne druhy využitia. LIRKIS CAVE disponuje komplexnejším tvarom, ktorý je unikátny pre pohltenie používateľa v trojrozmernom prostredí scény. Hlavnou myšlienkou pri realizovaní scén bola predstava využívania dostupných metodík podporovaných systémom pre zdokonalenie vnemu na vyššej úrovni používateľa z pohľadu grafického spracovania scény, dynamiky scény, doplnenia rozšírenia spätnej väzby. V dôsledku dosahovania efektivej scény na úrovni blízkej reálnemu vnemu boli aplikované pečené textúry trojrozmerných modelov so zahrnutím vlastností materiálov a svetelných podmienok, realizácia skybox prostredí pre docielenie nižších nárokov na výpočtový systém a eliminovaní prvkov komplexnej scény. Rozšírením externej aplikácie pre podporu ovládania scény prostredníctvom zariadenia Gamepad bola doplnená funkcionálna pre podporu spätnej väzby pomocou vibrácií ovládacieho zariadenia s možnosťou sledovania udalostí vo virtuálnom prostredí scény.

4.3.1. Aplikácia pečenia textúr

Vytváranie zložitých trojrozmerných modelov s komplexnou štruktúrou vplýva na nároky výpočtového systému. Používaním scénických svetiel obsiahnutých v SuperEngine prejavovali scény umelý výzor a časté chyby v generovaní tieňov opísané v predošlých kapitolách. Dôležité faktory pre dosahovanie reálnejšieho vnemu v trojrozmernom prostredí zohľadňujú odrazy plôch objektov, svetelnú efektívnu scénu, dynamické prostredie. Častou komplikáciou pri vytváraní scény v prostredí SuperEngine bol nárast nárokov na výpočtový systém v prípade používania svetelných prvkov s generovaním tieňov v reálnom čase. Rovnaké boli výsledky v dosahovaní reálnejšieho vnemu pri odrazoch svetelných lúčov alebo absencia zobrazenia zdroja svetelného prvku. V kombinácií svetelných prvkov a zložitých trojrozmerných modelov z pohľadu štruktúry, systém virtuálnej jaskyne prejavoval výrazné latencie vykresľovania viditeľné voľným okom používateľa. Pri aplikovaní transformácií objektov v trojrozmernom prostredí sa nároky znásobovali a reakcie systému boli oneskorené. Myšlienka vytvorenia scény s reálnym vnemom pre používateľa bola v uvedenom prípade náročná, nerealizovateľná. V dôsledku vytvorenia prostredia blízkeho realite sa uplatnilo použitie metódy pečenia textúr objektov s generovaním vlastností materiálov [37]. Pečenie textúr reprezentuje postup tvorby textúry s obsahom materiálu a svetelných prvkov scény, trvalým tieňom alebo odleskom povrchu objektu [Obr.31]. Postup pri vytváraní objektov s pečenými textúrami predstavuje vytvorenie objektov v modelovacom nástroji, aplikáciu

svetelných prvkov, materiálov na plochy objektov, vytvorenie siete modelov pre mapovanie textúry a záverečné renderovanie konečných textúr, ktoré sú v konečnom dôsledku nanášané na povrch objektu. Postupy môžu byť ovplyvnené podľa typu tvorenej scény, základný postup realizuje ako prvé modelovanie, druhé je realizovanie svetelných podmienok a tretie pečenie textúry priamo na navrhnutú sieť modelu. Scéna s aplikovanými pečenými textúrami prejavuje z pohľadu efektov statické prostredie, bez zmeny vlastností svetelných prvkov alebo generovania tieňov pri pohybe používateľa alebo zmeny pohľadu kamery.



Obr. 31 Scéna v prostredí SuperEngine s pečenými textúrami.

Spomínaný faktor je v skutočnosti nevýhodou budovania efektovo dynamickej scény. Z pohľadu realizácie pre dosiahnutie reálneho vnemu je aplikovanie pečených textúr nenáročné na nároky výpočtového systému v reálnom čase, dosahuje trvalý efekt pre používateľa. Pri vytváraní scén s pečenými textúrami sa v systéme virtuálnej jaskyne na Technickej Univerzite v Košiciach nepreukázali latencie systému, ktoré by spomaľovali jeho reakcie alebo vykazovali neschopnosť interakcií s používateľom, prípadne spôsobovali vysoké nároky na výkon výpočtového systému. Testované scény preukázali schopnosť systému aplikovať zložité štruktúry objektov s kombinovaním pečených textúr, v dôsledku ktorých scéna prejavovala výrazne lepšie hodnotenie z pohľadu používateľa pri dosahovaní podoby blízkej realite alebo vzbuďení dojmov pri používaní zariadenia OptiTrack.

4.3.2. Dosadenie Skybox objektov

Scény aplikované v systéme SuperEngine obsahovali pred testovaním v rámci diplomového projektu len jednoduché typy pozadí s typom jednej alebo prelínania viacerých farieb, prípadne jednoduchým prostredím reprezentovaným oblakmi. Z pohľadu dynamických vlastností bolo potrebné scénu obohatiť o jednotlivé prvky dynamiky a vytvoriť pre používateľa

reálnejší dojem vnímania okolitého prostredia. V dôsledku využívania transformácií na báze rotácií bolo vytvorených niekoľko vrstiev reprezentujúcich atmosféru prostredia virtuálnej scény. Najčastejším spôsobom realizácie prostredia je aplikácia jednoduchej panoramatickej fotografie na model gule, ktorej povrch je invertovaný smerom do vnútra, bez deformácií plôch alebo vytvárania dier modelu. V princípe je na povrch gule nanosená textúra zodpovedajúca vrstve atmosféry alebo druhu prostredia prostredníctvom fotografie, prípadne grafickej štruktúry reprezentujúcej konkrétny materiál. Skybox je možné používať nielen prostredníctvom atmosféry v okolí scény, rovnako je možná aplikácia pre realizáciu komplexnejšieho prostredia s vyššími nárokmi na výkon výpočtového systému. V prípade realizácie diplomového projektu bolo aplikovaných niekoľko scén s rôznymi typmi skybox prvkov pre rozšírenie okolitého prostredia scény. Z pohľadu časovej náročnosti je príprava prostredia skybox jednoduchá, závisí samozrejme od modelovacieho nástroja, ktorého vlastnosti predurčujú časové ohodnotenie tvorby prostredia. Rovnako je časová náročnosť závislá od skúseností osoby vytvárajúcej model prostredia. Aplikáciou prostredí skybox bol dosiahnutý reálnejší vnem pri simulovaní trojrozmerných scén. Výhodou je nízka náročnosť na výkon výpočtového systému a rovnako nižšie časové náklady oproti statickému vytváraniu rovnakého prostredia prostredníctvom vytvárania detailných modelov okolia. Spôsobom aplikácie skybox prostredí je možné dosadzovať objekty do pohľadu reálneho sveta, s presvedčením používateľa o skutočných rozmeroch trojrozmerného objektu v prostredí reálneho sveta.

4.3.3. Rozšírenie spätnej väzby

Myšlienka realizácie spätnej väzby rozšírila systém virtuálnej jaskyne a ďalšiu z funkcionalít pri simulácii scény trojrozmerného prostredia. Predošlý stav systému dovoľoval používateľovi využívať spätnú väzbu len na úrovni výpisov, prípadne naprogramovania výpisov údajov v okne konzoly Java Console. Nevýhodou pri interakciách systému a používateľa sú situácie, v ktorých je dôležité používateľovi preniesť spätnú väzbu viazanú na fyzický objekt reálneho sveta. Bežné simulačné zariadenia umožňujú využívanie rôznych spätných väzieb prostredníctvom systémov viazaných v prostredí používateľa a druhu aplikovanej simulácie. V prostrediach simulácie strojov je časté využívanie tlakového odporu prenášaného do ovládacieho zariadenia pripojeného do prostredia virtuálnej jaskyne. Príkladné sú páky v simulačných leteckých zariadeniach alebo žeriavoch, ktoré vytvárajú oproti používateľovi odpor v prípade, ak vo virtuálnom prostredí nastáva podobná situácia blokovania ovládacieho zariadenia. Využívaním externej aplikácie pre ovládanie scén prostredníctvom zariadenia Gamepad bola doplnená funkcionalita scén so zasielaním vibračných signálov používateľovi v prípade vybranej situácie v prostredí virtuálnej scény. Externá aplikácia sprostredkúva signály

medzi skriptom scény a Gamepadom, ktoré sú rozdelené podľa výstupných parametrov pri ovládaní trojrozmernej scény. V prípade spätnej väzby pomocou vibrácií, skript scény zasiela údaj o vibračnom signáli externej aplikácii zariadenia Gamepad. Externá aplikácia údaj eviduje a zasiela signál pre vykonanie vibračnej reakcie do zariadenia Gamepad, ktoré spúšťa vibračné motory. V uvedenom prípade môže používateľ systému pociťovať reálnejší dojem pri vytváraní interakcie so systémom, rovnako byť upozornený intuitívnejšou formou miesto výpisov alebo zobrazovaní údajov na výstupných grafických zariadeniach. Výhodou prenášania spätnej väzby do reálneho prostredia s viazaním na fyzický objekt je pohltenie používateľa v scéne a vytváranie intenzívnejšej väzby na úrovni komunikácie systému s používateľom.

5. Vyhodnotenie aplikácie systému SuperEngine

Aplikácia scén v LIRKIS CAVE so systémom SuperEngine od spoločnosti Slovakia SuperComputers bola využívaná v záujme overiť schopnosti systému a vizualizačného jadra. Úlohou kapitoly je zhodnotiť jednotlivé faktory pôsobiace na plynulosť vizualizácie scény v systéme LIRKIS CAVE. Scéna sa v počiatočných fázach zameriavala na využívanie dostupných prostriedkov vizualizačného jadra a predošlých ovládacích zariadení. Postupom riešenia bol systém LIRKIS CAVE obohatený o využitie inovatívnejších metodík v tvorbe reálnejšie pôsobiacich scén a rozšírenia ovládacích zariadení o Joystick a Gamepad. Náročnosť scén ovplyvňovali rôzne faktory, ktoré boli vyhodnotené s porovnaním náročnosti používaných štruktúr a ich vplyvom na výskyt latencií pri vizualizovaní scény. Rovnako boli zohľadnené situácie pre zistenie najčastejších vplyvov spôsobujúcich výpadky systému SuperEngine. Konečné hodnotenie prináša výsledky vplyvu grafických štruktúr na plynulosť vizualizácie scény. Porovnaním výsledkov experimentálnej časti odporúča používanie formátov grafických štruktúr pre optimalizovanie scény bez výskytu latencií.

5.1. Porovnanie výsledných funkcionalít

Systémy virtuálnych jaskýň disponujú rôznou škálou funkcionalít dostupných pre realizáciu scény v trojrozmernom prostredí, rovnako možnosťami interakcie používateľa so systémom. Funkcionality systému LIRKIS CAVE boli obmedzené z hľadiska dosahovania reálnejšieho vnemu pri vizualizovaní scén alebo rozšírením vstupných zariadení systému. Aplikovaním rôznych typov scén boli výrazne preukázané nedostatky samotného vizualizačného jadra, ktoré absentuje funkciami opísanými v predošlých kapitolách. Pre výrazne lepšie dosahovanie realistických scén je nutné používanie niektorých z metodík aplikácie textúr s využitím externého modelovacieho nástroja. Samotné rozšírenie funkcionalít bolo obohatené a aktualizované o knižnice systému SuperEngine pre zabezpečenie riadenia scény prostredníctvom

využívania dotykových tlačidiel a prvkov používateľského rozhrania, obsiahnutého v module JavaConsole. Rozšírenie uvedených funkcionalít docielilo nárast uplatnenia trojrozmerných scén a poskytlo dosahovanie reálnejšieho vnemu pri používaní systému používateľom. Prínosom v oblasti administrácie LIRKIS CAVE je rovnako aktualizované riadenie systému, ktoré umožňuje používateľovi riadiť jednotlivé procesy v SuperEngine pomocou JavaConsole. Výhody späté s aplikáciou nových funkcionalít predstavujú uľahčenie práce so systémom, vyššiu úroveň ovládania scénických objektov, riadenie systému z oveľa profesionálnejšej stránky.

5.1.1. Porovnanie vplyvu zložitosti 3D modelov

V procese overovania vlastností a reakcií systému bolo potrebné zhodnotiť vplyv náročnosti 3D modelov aplikovaných v scénach [Tab.1]. Dôvod predstavoval výskyt latencií systému a chyby zobrazovania v používaní rozdielnych formátov, rozdielnych zložitostí. Výsledky porovnávania zložitostí 3D modelov preukazovali výrazné rozdiely. Využívané formáty štruktúr predstavovali typy:

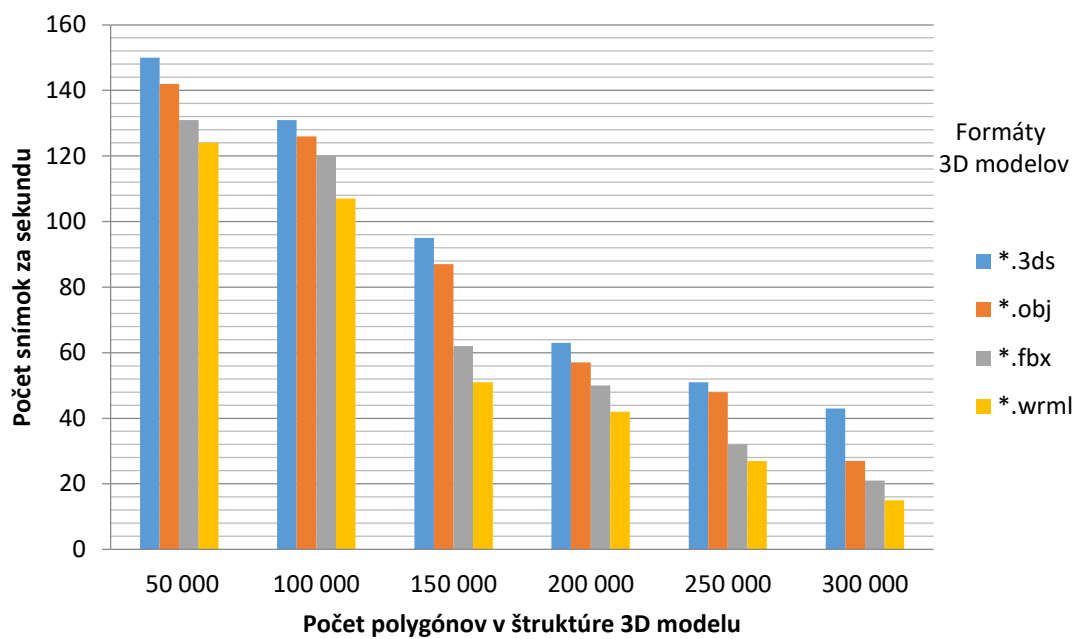
- *.3ds,
- *.obj,
- *.fbx,
- *.wrml.

Jednotlivé formáty boli používané v rovnakých prípadoch scén, s rovnakým počtom polygónov obsiahnutých v trojrozmernom modeli. Dôležité je hodnotenie optimálneho formátu používaného modelu, jeho kvality, detailov. Vysoké latencie predstavujú výrazné oneskorenie systému z pohľadu používateľa, napríklad viditeľne spomalené reakcie systému, výpadky scény, sekacie bez plynulej vizualizácie. Aby bolo možné latencie zdôrazniť, aplikovala sa miera pre sledovanie plynulosti vizualizácie scény, počtu snímok za sekundu (FPS). Porovnanie využívalo priemernú hranicu ľudského vnímania s počtom 24 snímok za sekundu.

Priemerný počet polygónov	50 000	100 000	150 000	200 000	250 000	300 000
Formát modelu						
*.3ds	150	131	95	63	51	43
*.obj	142	126	87	57	48	27
*.fbx	131	120	62	50	32	21
*.wrml	124	107	51	42	27	15

Tab. 1 Porovnanie FPS vplyvu formátov 3D modelov.

Výsledok zložitosti štruktúr preukázal najvhodnejšie používanie formátov *.3ds, ktoré v každom z prípadov prinášalo najlepšie hodnotenie [Obr.32]. Druhým z formátov *.obj je rovnako použiteľným s menšími rozdielmi oproti formátu *.3ds. Výrazné rozdiely predstavovali formáty štruktúr *.fbx a *.wrlm, ktoré spôsobovali pri vyššom počte polygónov výraznejšie spomalenie vizualizácie scény. Z pohľadu optimalizácie je formát *.3ds považovaný za najvhodnejší.



Obr. 32 Vplyv zložitosti štruktúr 3D modelov na plynulosť vizualizácie scény.

5.1.2. Porovnanie vplyvu textúr

Aplikovaním textúr viazaných na štruktúru 3D modelov bola hodnotená plynulosť vizualizácie scény s použitím rovnakého meradla, počtu snímok za sekundu (FPS). V záujme testovania použiteľnosti textúr boli jednotlivé textúry použité v rôznych formátoch a rozlíšeniach. Ako vzorka boli používané 3D modely s rozdielnymi štruktúrami formátov ale rovnakým počtom polygónov. Zámerom porovnávania rozlíšenia textúr a ich formátov na parametricky rovnakých vzorkách modelov bolo vyhodnotenie najvhodnejšej optimalizácie aplikovania textúr v SuperEngine.

Používané boli uvedené formáty textúr :

- *.bmp,
- *.jpg,
- *.png.

Z pohľadu veľkosti rozlíšenia textúr boli aplikované uvedené hodnoty:

- 768 x 768 pixelov,
- 1024 x 1024 pixelov,
- 2048 x 2048 pixel.

Každé z meraní bolo aplikované v prostredí rovnakej scény s rozdielnou štruktúrou trojrozmerných modelov a používaných textúr. Prvé meranie [Tab.2] prebehlo aplikovaním formátov textúr s uvedeným rozlíšením. Pre možnosť porovnávania vplyvu formátu textúry bola určená vzorka modelu so štruktúrou *.3ds a priemerným počtom polygónov 100 000. Sledovanie používalo mieru počtu snímok obrázkov za jednu sekundu.

Rozlíšenie textúry v pixeloch	768 x768	1024 x 1024	2048x2048
Formát textúry			
*.bmp	102	90	61
*.jpg	125	115	98
*.png	114	106	90

Tab. 2 Porovnanie vplyvu textúr s formátom modelu *.3ds .

Druhé meranie [Tab.3] predstavovalo aplikovanie textúr na odlišnom type formátu 3D modelu. Vybraný formát predstavoval štruktúru *.obj s rovnakou zložitosťou priemerného počtu polygónov na hranici s hodnotou 100 000.

Rozlíšenie textúry v pixeloch	768 x768	1024 x 1024	2048x2048
Formát textúry			
*.bmp	94	79	52
*.jpg	115	107	5
*.png	109	96	62

Tab. 3 Porovnanie vplyvu textúr s formátom modelu *.obj .

V predošlých meraniach boli viditeľné rozdiely medzi jednotlivými formátmi a rozlíšením použitých textúr a modelov. Nasledovalo vykonanie ďalších meraní využitím odlišnej štruktúry trojrozmerného modelu [Tab.4]. Tretím meraním bola uplatnená aplikácia textúr na štruktúre 3D modelu s formátom *.fbx s priemerným počtom polygónov 100 000.

Rozlíšenie textúry v pixeloch	768 x768	1024 x 1024	2048x2048
Formát textúry			
*.bmp	85	67	40
*.jpg	108	93	80
*.png	101	87	51

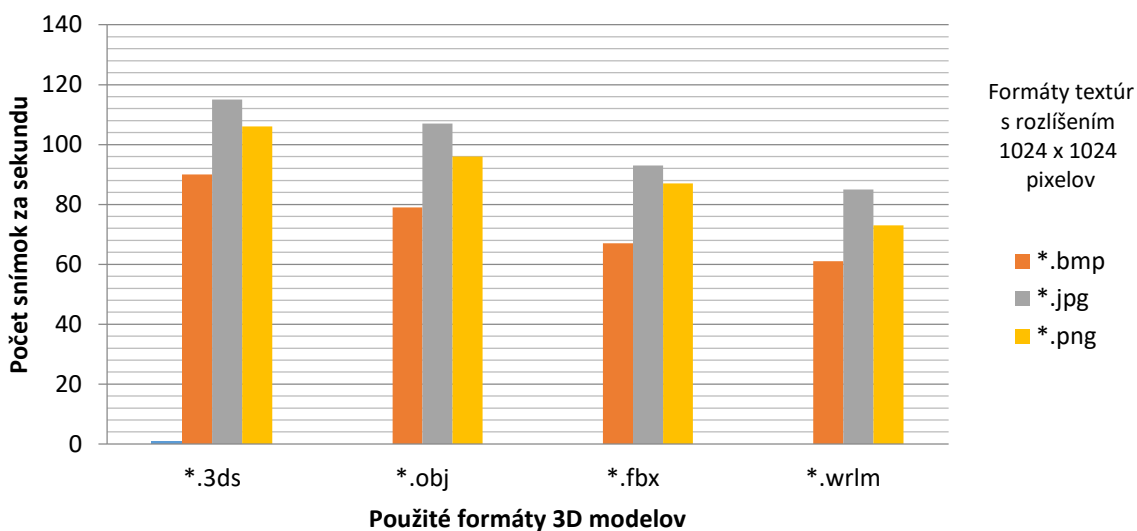
Tab. 4 Porovnanie vplyvu textúr s formátom modelu *.fbx .

Vykonanie posledného merania [Tab.5] obsahovalo použitie štruktúry 3D modelu vo formáte *.wrlm a zložitou priemernej počtu polygónov rovné hodnote 100 000.

Rozlíšenie textúry v pixeloch	768 x768	1024 x 1024	2048x2048
Formát textúry			
*.bmp	80	61	31
*.jpg	99	85	64
*.png	93	73	42

Tab. 5 Porovnanie vplyvu textúr s formátom modelu *.wrlm.

Výsledné hodnotenie vplyvu textúr preukázalo najvhodnejšie používanie formátu textúry *.jpg, reprezentovaného hodnotami v tabuľkách. Ostatné formáty textúr modelov predstavovali vyšší pokles zobrazovaných snímok za sekundu (FPS). Jednoznačným výsledkom pre odporúčanie optimalizácie scény v prípade priemernej rozlíšenia textúry 1024 x 1024 pixelov je používanie modelov s formátom *.3ds a textúrou reprezentovanou formátom *.jpg [Obr.33]. Formát *.jpg je komprimovaný, v prípade použitia na plochách objektov scény nie je výrazne viditeľný rozdiel v poklese kvality.



Obr. 33 Vplyv zložitosti štruktúr textúr na plynulosť vizualizácie scény.

5.2. Faktory ovplyvňujúce reakcie systému

Systém LIRKIS CAVE využíva sieťovú komunikáciu medzi hlavnou výpočtovou jednotkou a ostatnými výpočtovými jednotkami v počítačovom klastri. Prostredníctvom siete je riadené čítanie zo sieťového disku s obsahom scén a prijímanie konfiguračných informácií získavaných od hlavného počítača. Faktory vplyvajúce na chod systému sú v prípade LIRKIS CAVE rozdelené do kategórií, ktoré spôsobujú celkový pád systému, ovplyvňujú niektorý z modulov systému,

alebo odopierajú niektorú z funkcionalít. Najčastejším problémom je spôsob aplikovaný pri spúšťaní systému, ktorý môže prinášať odopieranie funkcionalít poskytovaných používateľovi. Rovnako je systém SuperEngine ovplyvnený nesprávnou konfiguráciou novo aplikovaných vstupných zariadení s ich nasadením. Vplyv v riadení scény závisí od správnosti implementácie skriptov, ktoré riešia funkcionalitu scény.

5.2.1. Vplyv postupu pri spúšťaní systému

V prípade LIRKIS CAVE je nutné dodržanie správneho postupu pri zapínaní systému. Nesprávne zapínanie zariadení prejavilo neschopnosť riadiaceho počítača ovládať inštancie SuperEngine na ostatných počítačoch v klastri, čo spôsobuje častý pád modulov Video Renderer Win 32.

Správny postup zapínania zariadení je uvedený v nasledujúcich bodoch :

1. Zapnutie počítača s aplikáciou Java Console.
2. Spustenie systému zapnutím klastra riadiaceho vykresľovanie scén.
3. Spustenie systému OptiTrack na hlavnom počítači klastra.
4. Spustenie simulácie scény v aplikácii Java Console.

Uvedené body sú dôležité pre zachovanie správneho narábania so systémom, v opačnom prípade systém nereagoval korektne a spôsoboval časté výpadky pri renderovaní scény. Ostatné faktory vplyvajú zo stránky konfigurácie alebo nesprávnej konštrukcie aplikovaného skriptu scény.

5.2.2. Vplyv konfigurácie vstupných zariadení

Vykonávaním konfigurácie zariadení boli v niektorých prípadoch objavené chyby spôsobujúce neschopnosť riadenia scény. Každý zo spôsobov konfigurácie prejavoval rôzne typy výpadkov systému sprevádzané buď neočakávaným správaním, odopretím rôznych funkcionalít alebo vypnutím a reštartovaním modulu Video Renderer Win32. V zámere odlíšiť druhy problémov konfigurácie bola vytvorená tabuľka [Tab.6] charakterizujúca faktory výpadkov s konkrétnym následkom a riešením problému.

Faktor	Následok	Riešenie
Nesprávne nakonfigurovaný typ zariadenia.	Výpadok modulu Control Center.	Kontrola konfigurácie typov.
Nesprávne nakonfigurovaný názov zariadenia.	Neschopnosť načítania scény.	Kontrola konfigurácie použitých názvov.
Nesprávne volané zariadenie v skripte scény.	Neschopnosť načítania scény.	Kontrola volaného zariadenia v skripte scény.

Tab. 6 Faktory problémov konfigurácie zariadení.

Nesprávne nakonfigurovaný typ zariadenia predstavoval očakávanie nesprávnych vstupov zo strany systému. Nie je možné napríklad vstupné zariadenie myš považovať za rovné zariadeniu klávesnice. Typovo sú odlišné, rovnako v prípade funkcií a komunikácie so systémom. Každý typ zariadenia je definovaný názvom, ktorý sa uloží v konfigurácii. Názov je samozrejme variabilný, ale v prípade použitia rovnakého názvu duplicitne nastáva problém. Ďalší z prípadov predstavuje volanie zariadenia v skripte scény, ktorého názov nie je evidovaný v konfigurácii. Všetky problémy sú riešiteľné spôsobom kontroly konfiguračných súborov alebo súborov skriptov s volaním korektného zariadenia.

5.2.3. Vplyv nesprávnej konštrukcie skriptu

V predošlých kapitolách boli uvádzané sekcie riadenia scény dôležité pri simulácii. Častý problém pri implementovaní scén spôsobovala absencia implementačného nástroja v systéme SuperEngine, podporujúca kontrolu syntaktických chýb. Nesprávny zápis premenných, volania metód, alebo hodnôt v metódach spôsoboval rôzne komplikácie prejavujúce sa pri spustení scény alebo jej simulácie [Tab.7]. So zámerom porovnania chýb bola aplikovaná tabuľka zobrazujúca vplyv nesprávne použitej konštrukcie alebo syntaxe skriptu pri vykonávaní vizualizácie scény.

Faktor	Následok	Riešenie
Nesprávne volaná metóda.	Neschopnosť spustenia scény.	Kontrola syntaxe zápisu.
Nesprávne zapísaný názov metódy.	Neschopnosť spustenia scény.	Kontrola syntaxe zápisu.
Nesprávne použité parametre.	Odopretie všetkých funkcií v scéne.	Kontrola parametrov metódy.
Nesprávne zapísané hodnoty parametrov metódy.	Odopretie všetkých funkcií v scéne.	Kontrola parametrov metódy a zápisu.
Použitie nedefinovaného znaku v kóde skriptu.	Neschopnosť spustenia scény.	Kontrola syntaxe zápisu.

Tab. 7 Faktory chýb v konštrukcii skriptu.

Každý z faktorov spôsobujúcich problémy je možné eliminovať pravidelnou kontrolou zápisu skriptu, používaných metód a ich parametrov, alebo nevyžiadaných znakov. Skutočnosťou neprítomnosti interného nástroja pre implementáciu skriptov scén je častá chybovosť z pohľadu zápisu kódu.

6. Záver

Výsledné aplikovanie scén v LIRKIS CAVE využívalo vizualizačný softvér SuperEngine prostredníctvom klastra počítačov zodpovedného za riadenie a zobrazovanie trojrozmerných prostredí. Výsledky testovania preukázali schopnosť systému riadenia scén s nižšími nárokmi na výpočtový systém, ktoré používajú jednoduchšiu zložitost' 3D objektov, primerané rozlíšenie textúr. Optimálnu záťaž predstavujú scény s priemerným počtom polygónov v rozsahu 100 až 200 tisíc pri aplikácii 3D modelov s formátom *.3ds. Používanie textúr bolo optimalizované s výsledným formátom *.jpg a maximálnym rozlíšením 1024 x 1024 pixelov. Dôvodom je verzia vizualizačného jadra OpenSG používaná z roku 2006, v ktorom boli výrazné absencie vizualizácie svetelných efektov a odrazov plôch podľa druhu materiálov objektov. Z opačného hľadiska je používanie systému výhodné prostredníctvom aplikácie metodík vytvárania trojrozmerných štruktúr a generovania ich podkladov pomocou pečenia textúr s využitím externého modelovacieho nástroja. Oproti ostatným vizualizačným jadram, obsahujúcim efekty pre prácu s časticovými systémami a svetelnými prvkami na vyššej úrovni, je nutné aplikovanie prostriedkov externého modelovacieho nástroja. Celkovo systém SuperEngine považujem za použiteľný, najčastejšie nevýhody predstavuje nestabilita a neaktuálnosť používaných knižníc Ruby 1.8.6. Z pohľadu rozšírenia bol systém obohatený o štandardné vstupy z klávesnice a myši, rovnako aplikovania zariadenia Joystick a Gamepad so spätnou väzbou. Systém virtuálnej jaskyne je možné doplniť ďalšími vstupnými zariadeniami prostredníctvom využitia hardvérových prvkov alebo aplikácií komunikujúcich so skriptom scény. Výsledným pohľadom je systém aplikovateľný pre vizualizácie jednoduchých scén a dynamických vlastností objektov bez vytvárania náročných štruktúr prostredia. Budúcnosť využívania systému je ohrozená vývojom moderných vizualizačných jadier s prevládajúcim počtom vizuálnych efektov.

Zoznam použitej literatúry

- [1]. BURDEA, Grigore C. a Philippe COIFFET. *Virtual reality technology*. 2nd ed. Hoboken, N.J.: Wiley-interscience, c2003. ISBN 0-471-36089-9.
- [2]. GUTIÉRREZ ALONSO, Mario A., Frédéric. VEXO a Daniel. THALMANN. *Stepping into virtual reality*. London: Springer, c2008. ISBN 1848001169.
- [3]. TALABĀ, Doru. a Angelos. AMDITIS. *Product engineering: tools and methods based on virtual reality*. Dordrecht, The Netherlands: Springer, c2008. ISBN 9781402081996.
- [4]. SHERMAN, William R. *Understanding Virtual Reality: Interface, Application, and Design*. ISBN 9780080520094.
- [5]. MCMENEMY, Karen a Stuart FERGUSON. *a hitchhiker's guide to virtual reality*. Wellesley, Mass.: a k Peters, Ltd., c2007. ISBN 1568813031.
- [6]. *Slovakia Super Computers* [online]. [cit. 2016-10-27]. Dostupné z: <http://www.supercomputers.sk>
- [7]. *Ruby Programming Language* [online]. [cit. 2016-10-27]. Dostupné z: <https://www.ruby-lang.org/en/>
- [8]. *OpenSG* [online]. Dostupné z: <http://www.opensg.org/>
- [9]. FÜNFZIG, Christoph. *Spherical techniques and their applications in a scene graph system: collision detection and occlusion culling*. Braunschweig: Göttingen Cuvillier, 2006. ISBN 9783867271110 3867271119.
- [10]. BAO, Hujun. a Wei. HUA. *Real-time graphics rendering engine*. Hangzhou: Zhejiang Univ. Press, 2011. ISBN 3642183417.
- [11]. JERALD, Jason. *The VR book: human-centered design for virtual reality*. First edition. San Rafael, CA: Morgan & Claypool, 2016. ACM book series. ISBN 978-1-97000-115-0.
- [12]. SOBOTA Branislav, VAVREK Miloš: *Triangulation of 3D scanned Objects*, Computer Science and Technology Research Survey, Košice, October 2010, 1, Košice, DCI FEEI TU Košice, 2010, 4, pp. 162-168, ISBN 978-80-553-0515-8.
- [13]. BIRN, Jeremy. *Digital lighting and rendering*. Indianapolis: New Riders Publishing, c2000. ISBN 1-56205-954-8.
- [14]. QVORTRUP, Lars (ed.). *Virtual interaction: interaction in virtual inhabited 3D worlds*. London: Springer, c2001. ISBN 1-85233-331-6.
- [15]. CRAIG, Alan B. *Developing Virtual Reality Applications: Foundations of Effective Design*. ISBN 9780123749437.

- [16]. THOMAS, David a Andrew HUNT. *Programming Ruby: the pragmatic programmer's guide*. Přeložil Yukihiro MATSUMOTO. Boston: Addison-Wesley, c2001. ISBN 0-201-71089-7.
- [17]. FORSYTH, David, Philip TORR a Andrew ZISSERMAN (eds.). *Computer vision - ECCV 2008: 10th European Conference on Computer Vision : Marseille, France, October 12-18, 2008 : proceedings*. Berlin: Springer, c2008. Lecture notes in computer science, 5302. ISBN 978-3-540-88681-5.
- [18]. 3DCONNEXION [online]. [cit. 2016-11-10]. Dostupné z: <http://www.3dconnexion.eu/products/spacemouse.html>
- [19]. THALES [online]. [cit. 2017-11-10]. Dostupné z: <http://www.intersense.com/>
- [20]. OptiTrack - Motion Capture Systems [online]. [cit. 2017-11-10]. Dostupné z: <http://optitrack.com/>
- [21]. KEMENY, Andras, Daniel J. SANDIN a Thomas A. DEFANTI. From driving simulation to virtual reality. *Proceedings of the 2014 Virtual Reality International Conference on - VRIC '14*. New York, New York, USA: ACM Press, 2014, , 1-5. DOI: 10.1145/2617841.2620721. ISBN 9781450326261.
- [22]. WRIGHT, Jonathan. Behind The Scenes at The Caterpillar Technical Center. In: *Peoriamagazines* [online]. [cit. 2016-12-2]. Dostupné z: <http://www.peoriamagazines.com/ibi/2012/may/behind-scenes-caterpillar-technical-center>
- [23]. STERN, Ch., H. NOSER, J. WEISSMANN a P. STUCKI. Application scenarios for scientific visualization and virtual reality using a CAVE infrastructure. *Proceedings of the workshop on Virtual environments 2003 - EGVE '03*. New York, New York, USA: ACM Press, 2003, , 319-320. DOI: 10.1145/769953.769997. ISBN 3905673002
- [24]. MANDAYAM, Shreekanth. Virtual Reality Environment at Rowan University. In: *South Jersey Technology Park* [online]. [cit. 2016-12-3]. Dostupné z: <http://sjtechpark.org/the-master-plan/the-cave/>
- [25]. SATAVA, Richard M., Eric BECKER, Fillia MAKEDON a P. STUCKI. Virtual reality surgical simulator. *Surgical Endoscopy*. New York, New York, USA: ACM Press, 1993, 7(3), 203-205. DOI: 10.1007/BF00594110. ISBN 9781450319737. ISSN 0930-2794
- [26]. GIESER, Shawn N., Eric BECKER, Fillia MAKEDON a P. STUCKI. Using CAVE in physical rehabilitation exercises for rheumatoid arthritis. *Proceedings of the 6th International Conference on Pervasive Technologies Related to Assistive Environments - PETRA '13*. New

- York, New York, USA: ACM Press, 2013, , 1-4. DOI: 10.1145/2504335.2504367. ISBN 9781450319737.
- [27]. MONIER, JEAN-FRANCOIS. VirTeaSy Surgery. In: *The Star* [online]. [cit. 2016-12-10]. Dostupné z: <https://www.thestar.com/business/2016/04/10/standing-up-for-the-little-guy.html>
- [28]. BATES, Joseph, Eric BECKER, Fillia MAKEDON a P. STUCKI. Virtual Reality, Art, and Entertainment. *Presence: Teleoperators and Virtual Environments*. New York, New York, USA: ACM Press, 1992, 1(1), 133-138. DOI: 10.1162/pres.1992.1.1.133. ISBN 9781450319737. ISSN 1054-7460.
- [29]. CORNELIS, Nico, Bastian LEIBE, Kurt CORNELIS a Luc VAN GOOL. 3D Urban Scene Modeling Integrating Recognition and Reconstruction. *International Journal of Computer Vision*. New York, New York, USA: ACM Press, 2008, 78(2-3), 121-141. DOI: 10.1007/s11263-007-0081-9. ISBN 9781450319737. ISSN 0920-5691.
- [30]. SINGH, Karan, Ravin BALAKRISHNAN, Kurt CORNELIS a Luc VAN GOOL. Visualizing 3D scenes using non-linear projections and data mining of previous camera movements. *Proceedings of the 3rd international conference on Computer graphics, virtual reality, visualisation and interaction in Africa - AFRIGRAPH '04*. New York, New York, USA: ACM Press, 2004, 78(2-3), 41-. DOI: 10.1145/1029949.1029956. ISBN 1581138636. ISSN 0920-5691.
- [31]. BIRN, Jeremy. *Digital lighting and rendering*. Indianapolis: New Riders Publishing, c2000. ISBN 1-56205-954-8.
- [32]. ISIDORO, John, Pedro SANDER. Animated skybox rendering and lighting techniques. In *ACM SIGGRAPH 2006 Courses (SIGGRAPH '06)*. ACM, New York, NY, USA, 19-22. DOI: 10.1145/1185657.1185827. ISBN:1-59593-364-6.
- [33]. Autodesk. [online]. [cit. 2016-12-16]. Dostupné z: <http://www.autodesk.com/>
- [34]. *Joystick* [online]. [cit. 2017-2-10]. Dostupné z: <https://en.wikipedia.org/wiki/Joystick>
- [35]. *SharpDX* [online]. [cit. 2017-2-10]. Dostupné z: <http://sharpdx.org/>
- [36]. *Gamepad* [online]. [cit. 2017-2-11]. Dostupné z: <https://en.wikipedia.org/wiki/Gamepad>
- [37]. *Polycount Forum: Texture Baking* [online]. [cit. 2017-2-18]. Dostupné z: http://wiki.polycount.com/wiki/Texture_Baking

Prílohy

- Príloha A: DVD médium – diplomová práca v elektronickej podobe, prílohy v elektronickej podobe.
- Príloha B: Používateľská príručka
- Príloha C: Systémová príručka